10/85
Slovenska revija za računalništvo in prosti čas
200 din

BIT

»Pametni« stroji ušli iz rok posvečenih

12 strani programov za vaš računalnik

VELIKA NAGRADNA IGRA

Kako lahko dobite s pomočjo kasete YAHTZEE, z nekoliko strateškega predvidevanja in ne preveč igranja:

1. **TIPKOVNICO INES**
2. **JOYPEN**
3. **IGRALNO PALICO**
4. **VMESNIK ZA SVETLOBNO PERO**
5. **5 RAČUNALNIŠKIH KASET v vrednosti 8.000 din**

In kaj morate storiti?

1. **Imeti morate kasete YAHTZEE in**
2. **prebrati navodila, ki so naslednja:**

YAHTZEE je stara japonska igra s petimi kockami, podobna pokru. Za Spectrum je prirejena, tako da lahko sodeluje deset igralcev, med njimi tudi vaš Spectrum.

Vsak igralec ima svojo tabelo s trinajstimi rubrikami, v katero vpisuje svoje točke. Igralec lahko trikrat vrže kocke. Pri drugem in tretjem poskusu lahko izbira kocke, ki jih želi ponovno vreči. Tako igralec skuša dobiti izbor očesc na kockah in vpisati ustrezno število točk v svojo tabelo.

Polja v tabeli so naslednja:
najprej so to polja: ENICE, DVOJKE, TROJKE, ŠTIRICE, PETICE in ŠESTICE, kamor vpišemo vsote kock z ustreznim številom očesc.

Nato sledijo polja:
TROJICA — tri enake kocke, kamor vpišemo vsoto vseh kock
ČETVERICA — štiri enake kocke, kamor vpišemo vsoto vseh kock
POLNO — dve enaki in tri enake kocke, kamor vpišemo 25 kock
MALI NIZ — zaporedje štirih kock, kamor vpišemo 30 točk
VELIKI NIZ — zaporedje petih kock, oziroma lestvica, kamor vpišemo 40 točk
YAHTZEE — pet enakih kock, kamor vpišemo 50 točk
REZERVA — izbor kock ni predpisan. Vpišemo vsoto vseh kock

Če je polje za določen izbor kock že polno ali izbor kock ne ustreza nobenemu polju v tabeli, lahko igralec vpiš točke v polje REZERVA. Če pa je tudi to polje že polno, mora vpisati ničlo v eno izmed praznih polj v tabeli.

Igralec, ki zbere v prvih šestih poljih ENICE, OVOJKE, TROJKE, ŠTIRICE, PETICE in SESTICE vsaj 63 točk, dobi nagrado — BONUS, ki prinese 35 točk. Ko vsi igralci popolnijo svoje tabele, je igre konec. Zmagal je igralec, ki je zbral največ točk. Vsakemu številu točk ustreza šifra, sestavljena iz petih črk. Ta šifra je pomembna za nagradno igro ZOTKS in revije BIT.

NOVE KASETE IN PRIPOMOČKI

Toliko kaset in drugih računalniških pripomočkov zmoreta le revija BIT in Zveza organizacij za telesno kulturo Slovenije ZOTKS:

1. **Kaseta YAHTZEE**	1.300 din
2. **Zemljepis za 6., 7. in 8. razred osnovne šole**	1.200 din
3. **Energetske osnove SFRJ**	1.200 din
4. **Radioamaterska kaseta CW MORSE**	1.300 din
5. **Mavrični grafikon**	1.300 din
6. **Programi za JOYPEN — kaseta s 4 programi LIGHT-PEN — svetlobno pero — vmesnik**	18.000 din
7. **PODATKOVNA BAZA** (prava posebna domača kaseta za C-64 COMMODORE)	1.300 din
8. **Kaseta z 2 igrama** in vmesnikom, ki se lahko programira	15.000 din

Vse kasete in pripomočke lahko naročite z objavljeno naročilnico. Vi ste naši ustvarjalci, zakaj ne bi skrbeli za vas!

BIT — ZOTKS

oblikovanje **bep**

Tokrat domače presenečenje

Supermikroračunalnik Triglav: podobnega še ni


Nov izdelek Iskre-Delte je oblikovan izrazito sodobno

Pred dnevi je bila v Cankarjevem domu predstavljena nova družina 16/32-bitnih supermikroračunalnikov našega vodilnega proizvajalca, ISKRE-DELTE. Ime TRIGLAV prihaja od treh mikroprocesorjev, ki so za sistem najbolj značilni — Motorola 68010, Intel iAPX 80286 in Digital Equipment J 11; poleg njih je na voljo, ljubiteljem operacijskega sistema CP/M v tolažbo, še Zilogov Z 80 B.

Kot je na lepo pripravljeni tiskovni konferenci povedal direktor ing. Janez Škrubej, gre za tehnološki preboj; noben svetovni gigant doslej še ni predstavil mikroračunalnika s takimi zmogljivostmi.

Posebnost stroja je namreč v tem, da ima standardno VME vodilo z devetimi vtičnicami za module formata dvojne Evrope. Teh modulov je pa pri družini Triglav več vrst — od takih s centralno procesno enoto (na enem modulu eden izmed zgoraj naštetih mikroprocesorjev, vsak lahko naslavlja do 16 Mb) in 256 Kb do 1 Mb pomnilnika, do spominskega modula (512 Kb ali 2 Mb), grafičnega modula (notranja ločljivost 1024 × 1024 pik, 16 barv izmed 256), serijskega komunikacijskega modula (6 RS 232 C kanalov) pa do modula s krmilnikom za trdi (10 — 40 Mb) in gibki disk (1 Mb). Zaslon, ki je lahko barven (480 × 640 pik) ali črnobel in tipkovnica, ki ima dodanih 19 funkcijskih tipk, sta oba VT 100 kompatibilna.

Skratka, računalnik lahko sestavimo npr. iz treh modulov s po enim procesorjem, iz modula s krmilnikom za diskovni enoti, grafičnega modula, komunikacijskega modula, pa še ostane prostor za tri spominske module s 6 Mb pomnilnika. Vrhunski dosežek nedvomno predstavlja dejstvo, da lahko več mikroprocesorjev hkrati teče, ne da bi drug drugega motili.

K takemu strojnemu razkošju seveda sodi tudi ustrezno veličastna programska podpora — na razpolago je kar 9 (devet) operacijskih sistemov: CP/M 68 K, UCSD-p, MS-DOS, Concurrent-DOS, UNIX, XENIX, RSX 114, RT 11 in DELTA/M. CP/M 68 K omogoča kompatibilnost z osebnim računalnikom ATARI 520 ST, MS DOS z IBM PC, XENIX s PC/AT, zadnji trije pa z DEC miniračunalniki serije PDP 11 oziroma DELTA 400, 644 in 800. Pri programskih jezikih so na voljo C, Fortran, Basic, Cobol, Pascal in različni zbirniki.

Zunanja podoba računalnika je delo studia MAK iz Kopra; design je izrazito sodoben, »industrijski« in, kot se na slikah lahko sami prepričate, res izviren.

Oglejmo si še cene in dobavne roke. Računalnike bodo konec leta 1985 začeli sestavljati v podružnici ISKRA-DELTA Computers na avstrijskem Koroškem, v Šentjakobu pri Podrožci. V podjetju bo z druževalo delo 25 strokovnjakov, ki bodo v Kranju izdelano osnovo dopolnjevali z na svetovnem trgu, predvsem na Japonskem, kupljenimi komponentami, diski in zasloni.

Računalnik, ki bo na tujem nosil ime TRIDENT, bodo izdelovali v treh izvedbah — kot grafično delovno postajo (za uporabo pri CAD/CAM sistemih), kot računalnik za procesno kontrolo (v industriji, pri robotih) in kot večuporabniški poslovni sistem. Prvi dve izvedbi naj bi bili gotovi do konca 1985, poslovna inačica pa v 1986. Izvozna cena grafične delovne postaje (en mikroprocesor, en operacijski sistem, en programski jezik, 20 Mb trdi disk, 1 Mb gibki disk, 512 Kb RAM, barvni zaslon) bo znašala okoli 7.000 ameriških dolarjev, ustrezna cena na domačem trgu pa 6.900.000 dinarjev. Prvo leto naj bi v Šentjakobu predvidoma izdelali 300 računalnikov in, če bo šla prodaja na tujem dobro, se bodo v letu 1986 pojavili prvi TRIGLAVI tudi na domačem trgu.

Kaj naj rečemo na koncu pod vse skupaj? ISKRI-DELTI in avtorjem stroja gre nedvomno vse priznanje. V tem, razvoju lastne pameti tako malo naklonjenem prostoru in času, smo tudi Slovenci napravili res sodoben, lasten in svetovljansko oblikovan računalnik.


Strojno razkošje: Triglav ima na voljo kar devet operacijskih sistemov

Primož Jakopin

V tej številki

Številka Bita, ki je pred vami, ima za polovico več programov kot je to bilo običajno doslej. Prilogo Zeleni Bit smo razširili na osnovi vaših številnih predlogov, pa tudi programov, ki nam jih vse pogosteje pošiljate. Poziv velja seveda še vnaprej!
Več pisem smo dobili tudi za rubriko Bit-borza, še posebej za sporočila »menjave-nasveti«, ki so brezplačna. Prav tu pa je očitno prišlo do majhnega nesporazuma, saj so nekateri bralci menili, da so brezplačna tudi sporočila za prodajo ali nakup. Takšne oglase seveda zaračunamo po ceni, ki je objavljena na priloženi dopisnici, sporočila za menjave ali nasvete pa bomo objavili brezplačno. To možnost še posebej omenjamo zaradi vse številnejših računalniških klubov, ki želijo čimveč stikov z lastniki istih računalnikov in je tudi to morda priložnost, da bi takšno komunikacijo okrepili.
Za številne bralce, to lahko vsaj razberemo iz pisem, je zanimiva tudi rubrika »Igre meseca«. Doslej smo ocenjevali le igre za spektrume, od prihodnje številke pa bomo podoben prikaz pripravljali tudi za commodorje.

BIT Slovenska revija za računalništvo in prosti čas

Izdaja: TOZD ČLP Ljubljanski dnevnik v sodelovanju z Zvezo organizacij za tehnično kulturo Slovenije in Tehnično založbo Ljubljana.
Odgovorni urednik: Robert Mecilošek, Redakcija: Tamara Lah, Boris Horvat, Gorazd Marinček, Dušan Kralj, Slobodan Rakočevič, Herman Savec, Sandi Sitar, Tomaž Skulj.
Glavni urednik Dnevnika: Milan Meden • Odgovorni urednik: Edo Glavič • Direktor TOZD: Drago Bitenc
Telefoni uredništva: 325-752 in 323-841. Biro za ekonomsko propagando 317-954 • Prodajno-naročninska služba: 325-261 • Reklamacije: 325-747 • Cena 200 dinarjev • Žiro račun pri SDK, Podružnica Ljubljana, št. 50100-603-41518. Oproščeno prometnega davka.
Tiska. Tiskarna Ljubljana

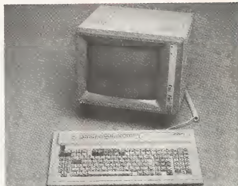
Commodore 64: najcenejša videotipkovnica

Videotekst je komunikacijsko sredstvo, ki se je razvilo sredi 70. let. Kdor ima televizijski sprejemnik, ki je prirejen za sprejem videoteksta, lahko na zaslonu prebere poročila, dobi potrebne informacije o prometnih sredstvih in drugih storitvah, v tujini pa tudi naroča v veleblagovnicah, rezervira mesto v letalu ali gledališču in vodi bančni račun. Tudi RTV Ljubljana je začela s poskusnimi oddajami. Za sprejem videoteksta moramo imeti primeren televizor ali poseben dekoder zanj, tipkovnico ali aparat za daljinsko upravljanje, modem za priključitev na telefon in telefon. Od leta 1983 je mednarodni standard za videotekst tako imenovani CEPT standard. Na vidotekstu je mogoče prikazati vse črke latinice, ima izboljšane grafične zmogljivosti in v eni sliki lahko izmed 4096 različnih barv izberemo 32 barv za barvo ozadja in 32 za barvo črk. Za sprejem in obdelavo podatkov iz tega komunikacijskega omrežja in za pošiljanje podatkov v videotekst omrežje uporabimo tudi prirejene osebne in hišne računalnike. Podatke, ki jih pošiljamo po mreži, lahko omejimo krajevno ali pa tako, da so dostopni le določenemu krogu ljudi, na primer kupcev ali sodelavcev.

Prek serijskega vmesnika RS32C lahko na terminal za videotekst priključimo skoraj vse osebne računalnike. Vse funkcije, ki jih sicer vklapljamo s pritiskom na tipke, opravlja računalnik. Računalnik lahko dela kot sprejemnik ali pa obdeluje sprejete podatke. Vendar pa moramo imeti v ta namen potrebno mehko opremo, ker moramo strani videoteksta najprej pretvoriti v datoteke, šele nato jih lahko shranimo in izpišemo na tiskalniku. Z računalnikom lahko pošiljamo sporočila in izpolnjujemo formularje. Strani vidoteksta lahko tudi editiramo in jih pošljemo drugemu računalniku. Commodore 64 spreobrnemo v terminal z alfanumerično tipkovnico s posebnim modulom. Z računalnikom pokličemo centralo, izberemo strani, pišemo odgovore, na koncu pa prekinemo zvezo. Za hitrejši pogovor lahko ukaze shranimo na disketi in jih kasneje spet uporabimo. Na disketi lahko shranimo do 100 strani videoteksta in jih nato v miru pregledujemo, ne da bi nam bilo treba še enkrat klicati centralo.

Samo pri dveh računalnikih je za dekoder uporabljena čista programska rešitev, pri profesionalnem osebnem računalniku Texas Instruments in pri profesionalnem prenosnem računalniku istega proizvajalca. Dodatna trda oprema za sprejem videoteksta (dekoder in zaslon) pri teh dveh računalnikih ni potrebna, zadosti je, da poženemo poseben program. Možna je tudi nadaljnja obdelava sprejetih podatkov.

Če hočemo sprejemati videotekst z IBM PC, XT ali AT moramo uporabiti poseben adapter. Grundig je za Ibmove računalnike izdelal poseben prikazovalnik s tremi načini delovanja: enobarvno, barvno in videotekst.



Rafis Btx terminal z vgrajenim dekoderjem in tipkovnico

Canon V20, Goldstar FC 200, Sanyo MPC-100

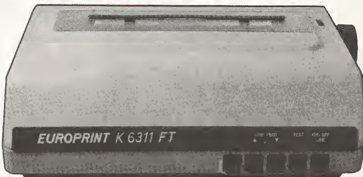


Naraščaj v družini MSX

Na trgu so trije novi MSX računalniki. Hardversko niso nič posebnega ali podobnega Yamahinemu Musik MSX ali Sonyjevemu Hit Bit. Canonova in Sanyova tipkovnica sta elegantni, Sanyova se srebrno svetlika na črni podlagi, Cannova je siva, sem in tja pa jo izbrano popestrijo oranžne oznake na tipkah. Goldstar daje malo drugačen optičen vtis, tipke so rdeče,

Velikost ne vpliva na hitrost

Že na prvi pogled daje matrični tiskalnik K 6311 vtis robustnosti. Ko dvignemo pokrov, vidimo, da smo imeli prav. Najprej opazimo precej nenavadno pisalno glavo. Zaradi zelo velikih navitij je skoraj dvakrat večja od običajnih. Vendar pa to ne vpliva kaj dosti na hitrost, s 100 znaki na sekundo pri dvosmernem pisanju se K 6311 ne loči od večine drugih tiskalnikov tega razreda. Znaki so sestavljeni iz matrike 9 × 7 pik in zato lepo pišejo tudi male črke in črke s podaljški (npr. j, g). Ima osem mednarodnih naborov znakov s po 95 znaki. Kot skoraj vse druge pa lahko tudi nabor znakov izbiramo le programsko.



Na sprednji strani ima pet tipkal, eno nima funkcije, druga pa vsako po dve. Delujejo kot normalna stikala in hkrati kot kontrolne lučke, ki opozarjajo na napake, konec papirja, itd.

Na žalost ne moremo nastaviti levega in desnega robu in preskoka prek konca lista. V vrstici izpisujeom 80, 100 ali 120 znakov. Pisava je normalna, široka ali kurzivna. V grafičnem načinu imamo 600 pik v vrsti.

Način priključitve (serijski ali vzporedni vmesnik) izbiramo s kartico, ki se nahaja v zadnjem delu tiskalnika. Po želji izberemo serijski ali Cetronics vmesnik. Možna je torej tudi naknadna preureditev. Vendar pa moramo pri tem spremeniti tudi mehko opremo, ki je v tiskalnik vgrajena v obliki modula. Tiskalnik K 6311 FT je v prodaji tudi v naših trgovinah.

modre in zelene. Po tej plati je očitno namenjen mlajšemu računalniškemu rodu, medtem ko bosta druga dva s preudarno rednostjo vzbujala zaupanje veteranskemu kadru.

Vsi trije msiksovci imajo svetlobno diodo, ki nam pove, da smo vklopili velike črke, Goldster pa ima še eno za tipko »Code«.

Canonova tipkovnica se zdi malo slabša in manj primerna za največje obremenitve kot Sanyova. Preslednica je precej majhna. Funkcijske tipke in smerne tipke pa so velike, kot se spodobi, celo večje kot pri kateremkoli drugem MSX računalniku.

Sanyova tipkovnica je odlična. Funkcijske tipke niso tako velike kot pri Canonu, vendar so navzlic temu primerne, posebej pa je treba pohvaliti lego smernih kazalcev.

Najbrž je med temi tremi računalniki še najslabša Goldstarjeva tipkovnica, ki ima nekaj šibkih točk. Tipki »Return« in »Shift« sta premajhni. Funkcijske tipke so dobre, smerne tipke premajhne. Za editiranje na zaslonu so zadosti velike, premajhne pa so za igranje hitrih iger.

Vsi trije računalniki imajo poleg priključkov, ki jih predpisuje standard, vmesnik Centronics za tiskalnik. Canon ima še en modulski konektor, ki se nahaja na levi strani okrova in je primeren za priključitev disketne enote. Goldstar in Sanyo pa imata namesto tega razširitveni konektor, tako da lahko priključimo periferne naprave, ki so opremljene z ustreznim vmesnikom.

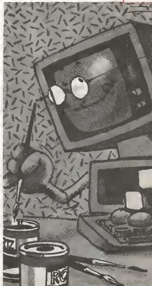
Reset stikalo ima le Sanyo. Canon ima vtičnico za veseli palici na sprednji strani. Goldstar in Sanyo imata vgrajen tudi varnostni sistem: če pomotoma vtaknemo modul v konektor pri vključenem računalniku, se računalnik sam izklopi.

Cena: okoli 200 funtov. Goldstar morda ne naredi najboljšega vtisa zaradi slabše tipkovnice, je pa zato najcenejši. Sanyo in Canon sta primerna za vse, ki morajo veliko pisati in ki hočejo imeti solidno tipkovnico.

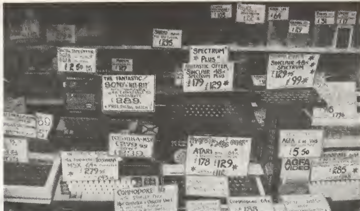
Kaj pa je novega pri MSX programih. Medtem ko je japonska poudba programov na razmeroma dragih ROM modulih precej skromna, je vse več programov iz Anglije.

Številne programske hiše, ki so doslej delale le za spectrume, so se preusmerile na Amstradov CPC 464 in MSX računalnike.

Poleg številnih igric so na kasetah na razpolago tudi Forth, disasembler in uporabniški programi, na primer priljubljeni Tasword II. Pri igricah pa je vse več priredite priljubljenih igric za spectruma, druščino si delajo Vili iz visoke družbe, Nogometni šef, in Grog s svojim maščevanjem:



V Angliji (in tudi drugje) je spet zabesnela cenovna vojna med vodilnimi, Sinclairjem, Acornom, Bbcjem in Commodorjem. Cena spectruma plus je padla na 129 britanskih funtov in stane toliko kot 64 KB Atari 800 XL. Da bi se izognil direktnemu spopadu z Atarijem, pa Sinclair prodaja komplet za razširitev navadnega spectruma 48 K v spectrum plus s tipkovnico, novim priročnikom za uporabo in demonstracijsko kaseto za borih 20 funtov. Tako dobljeni spectrum plus stane le 119 funtov. Commodore 64 stane 188 funtov in je torej precej dražji. Vojno pa so si napovedali le proizvajalci, med prodajalci vlada mir. Hišni računalniki stanejo skoraj v vseh trgovinah enako. Za nas je torej vseeno, ali ga kupimo v blagovnici, trgovini z računalniki ali trgovini z gramofoni, QL stane povsod 379 funtov.



Vsak dan nove cene — vendar nižje... Izložba ene od londonskih trgovin

Neusmiljena cenovna vojna

Angleški trg hišnih računalnikov je prezasičen

Na letošnji razstavi LET se je dobro odrezal Schneider oziroma Amstrad CPC 464, pri dodatni opremi pa se je najbolje odrezal Wafadrive. Za Sinclairjev QL so bili prikazani poleg razširitve pomnilnika različni disketni sistemi s kapaciteto do 1 MB za 320 do 400 funtov. Skromna je bila ponudba programov za QL: omembe vredni so bili le štirje: Psionov šah, asembler, Toolkit in Cash Trader). Močan program za igranje šaha z odlično tridimenzionalno grafiko stane okoli 20 funtov, asembler 40, Toolkit 25 in program za knjigovodstvo 65 funtov. Timex je skupaj s Profisoftom prikazal disketno enoto z operacijskim sistemom CP/M za spectrum in commodore 64, ne ve pa se, kdaj bo naprodaj. Mehka hiša OCP ponuja poslovne programe pod imenom SP--DOS Address Manager, Stock Manager in Finance Manager za mikrotračno enoto, Wafadrive, Beta disk in Timexov sistem. Kdaj jih bodo prodajali pri MK ni znano.



Kljub poceni plusovi tipkovnici ponuja Saga Systems dve izboljšani tipkovnici za spectruma. Sago 2 privijemo na spectrum odspodaj, ima 71 tipk in posebno številčno tipkovnico. Saga 3 pa vzame vase celo spectrumovo tiskano vezje in ima še deset funkcijskih tipk. Zato stane približno 70 funtov, Saga 2 pa le 60 funtov. Saga 1 je naprodaj že za 50 funtov. Saga GP pa je grafična tablica za IBM, Apple, Spectrum in Commodore, pripravljajo pa tudi izvedbe za BBC, MSX in Schneider-Amstrad.

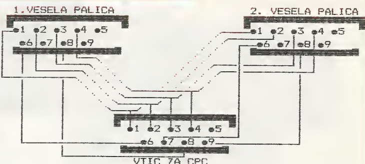
Kempston ponuja novo serijo vmesnikov oblikovanih v plusovem slogu: Centronics vmesnik za ca. 40 funtov, navadni vmesnik za igralno palico za 10 funtov in Interface 2, ki ima še en priključek več, za približno 20 funtov. Tudi AGF ima nove vmesnike za vesele palice. Odličen je programabilni vmesnik Protocol-4 za ca. 25 funtov. Datel Electronics ponuja za 50 funtov »sound interface« in za 30 funtov robotski vmesnik.

Cheetah in Currah prodajata sintetizator govora za spectrum in commodore 64. Kar se tiče iger, je le malo novega in izvirnega. Očitno manjka idej. Mogoče lahko svetujemo Kempstonu, naj izdela joystick interface za Sinclairjev elektromobil, tako da si voznik ne bo obrusil komolcev, ko bo obračal volan.

Priključek za dve palici

Odpravimo pomanjkljivost na Schneiderju CPC 464

Spajkalnik, malo »cina« in žice in že lahko na CPC 464 priključimo katerokoli veselo palico.



Kdo ve, zakaj je Schneider svojemu računalniku privoščil le en sam priključek za veselo palico, tako da lahko drugo priključimo le na Schneiderjevo igralno palico. Kdor se noče ločiti od stare dobre pomočnice, ki ga je izvlekla že iz marsikakšnih škripcev, naj si priskrbi 9-polni Atari vtič (takšnega, kakršen je na kablu igralne palice), dva moška vtiča in nekaj žice.
Na moška vtiča bomo priključili veseli palici, ženski »CPC vtič« pa bomo priključili na računalnik. Na sprednji strani vtičev so napisane številke posameznih kontaktov. V zgornji vrsti številke od 1 do 5, v spodnji od 6 do 9. Pri ženskih vtičih tečejo številke od leve na desno, pri moških od desne na levo. Vezavo kaže slika. Prve štiri kontakte z obeh konektorjev za igralni palici vežemo skupaj. Smer gibanja je izražena v številčni vrednosti, ki jo dobimo z ukazom JOY(0) za prvo oziroma JOY(1) za drugo veselo palico.

Združljivost mu ni neznana

Nixdorf CPC 8810/22



Nixdorf CPC 8810/22 je prenosljiv, kompakten, popolnoma združljiv z IBM PC. Ima vgrajen termični tiskalnik, ki v eno vrsto napiše 80 ali 132 znakov. V osnovnem okrovu, ki ga pokrijemo z IBM PC združljivo tipkovnico, je glavno vezje z razširitvenimi konektorji, serijski in vzporedni vmesnik, devetpalčni monitor jantarne barve in tiskalnik. V standardno opremo sodita dve 360 KB disketni enoti. Eno lahko zamenjamo z 10 MB trdim diskom. Napajalna enota je izdelana, tako da zlahka prenese tudi takšno razširitev sistema.

Če hočemo imeti stabilni delovni sistem, uporabimo razširitveno enoto, v kateri je prostor za dodatne IBM kompatibilne kartice, cenen risalnik in barvni prikazovalnik, velikosti 12 palcev po diagonali. Prenosljivi namizni računalnik, kakor Nixdorf imenuje svoj ponos, ima vse pomembne strojne komponente v enem samem okrovu, vključno s tiskalnikom. Zato so vsi kabli na mizi odveč. Težak je 15 kg in oblikovan, tako da najde prostor tudi pod sedežem v letalu.

Vgrajen serijski in vzporedni vmesnik omogoča zadostne možnosti priključitve. Za profesionalni trg, kateremu je računalnik namenjen, je zanimiv, saj stane kompletno opremljen sistem manj kot 7000 DM.

TEHNIČNI PODATKI

Mikroprocesor: 8088 4,77 MHz, konektor za koprocesor
Pomnilnik: standardna izvedba 256 KB, do 640 KB
Vmesnik: serijski: RS232C, vzporedni: Centronics, Razširitveni konektorji: 2, v dodatni enoti še 3
Zaslon: 9-palčni enobarvni (vgrajeni), 12-palčni barvni (zunanji)
Ločljivost: 640 × 200 enobarvni, 320 × 200 barvni, 25 vrst × 80 znakov ali 25 vrst × 40 znakov
Disketna enota: 2 enoti po 360 KB
Tiskalnik: Termični, 80 ali 132 znakov v vrsti
Nabor znakov: skladen z Epson MX 80, širina papirja maks. 216 mm
Operacijski sistem: MS-DOS 2.11
Mere: dolžina 470 mm, širina 335 mm, višina 210 mm

Za redukcijo se ne zmenijo



Prenosni termični pisalni stroji — tiskalniki za ZX spektrum

Že nekaj časa po Evropi prodajajo majhne, lahke, skratka prenosne električne pisalne stroje za vse, ki se nikjer ne morejo ali nočejo ločiti od svojega dela. Zmeraj so pripravljeni za pisanje, za redukcijo se ne zmenijo. Te pisalne stroje pa lahko uporabimo tudi kot tiskalnike za hišni računalnik in pri tem celo precej prihranimo. Oglejmo si tri prenosne termične pisalne stroje, ki jih zaradi vgrajenega vmesnika RS232 lahko priključimo na Spectrum, ki pa mora seveda imeti pod sabo vmesnik Interface I.

Najcenejši termični pisalni stroj je Brother EP 22, ki ga trenutno ponujajo precej pod priporočeno ceno (kar pomeni, da ni več najsodobnejši). Tako kot druga dva piše na zelo kvaliteten, satiniran papir, normalen ali termopapir. Če pišemo na normalen papir, potem potrebujemo poseben termični pisalni trak, ki pa ga lahko uporabimo le enkrat samkrat. Odtis je v obeh primerih enak, čeprav zaradi majhnega števila pik (5 × 7) ne posebno dober. Največja hitrost pisanja je 17 znakov na sekundo. V vrstico napiše do 75 znakov.

S stikalom za izbiro načina delovanja izbiramo med normalnim pisanjem, pisanjem v posebni pomnilnik in delovanjem kot tiskalnik. Pomnilnik shrani 1500 znakov (približno eno tipkano stran oziroma 27 vrstic) in ga lahko popravljamo na majhnem zaslonu iz tekočih kristalov. Ko ga priključimo na računalnik, moramo najprej izbrati hitrost prenosa. Na voljo imamo dve hitrosti, 75 in 300 baudov (znakov na sekundo). Število podatkovnih ali ustavljalnih bitov na žalost ne moremo spreminjati.

Kdor se je razvadil na navadnih tiskalnikih s kurzivo, širokimi in komprimiranimi črkami in proporcionalno pisavo, ta bo seveda razočaran. Na razpolago imamo samo tri krmilne ukaze: pomik na začetek vrste, pomik v novo vrsto in pomik nazaj ali po domače CR, LF in BS. Zato moramo besedilo dokončno oblikovati že pred izpisom.

Naslednik EP 22 je EP 44. Njegova tipkovnica je sicer na prvi pogled enaka, podrobnejši pregled pa pokaže vrsto razločkov. Predvsem pa so črke veliko lepše, kar je razumljivo, saj so sestavljene iz 24 × 18 pik. Kakovost izpisa torej ne zaostaja dosti za izpisom z marjetičnim tiskalnikom.

Pomnilnik EP 44 je skoraj dvakrat večji od EP 22. V njej je prostora za 3726 znakov in omogoča editiranje posameznih znakov ali celih vrstic. V pomnilniku lahko brišemo ali vanj vnašamo cele vrstice. Besedilo lahko tudi naknadno dodajamo. S smernimi tipkami se premikamo po besedilu sem in tja kakor hočemo. Pomnilnik za napisano besedilo je pomemben tudi zato, ker zaradi tehnike pisanja ni mogoče pisanje v kopiji.

S posebnim stikalom izbiramo med pisalnim strojem in tiskalnikom. Vendar pa tiskalnik pozabi, kakšen je bil kot pisalni stroj, na primer na nastavitev tabulatorja, desnega ali levega robu. S krmilnimi kodami pa tega na žalost ni mogoče popraviti. Kot tiskalnik namreč EP 44 razume le sedem kontrolnih kod. Omenimo najpomembnejše: pomik nazaj, vodoravni tabulator, pomik v novo vrsto in pomik na začetek vrste.

V primerjavi z drugima dvema pisalnima strojema imamo pri EP 44 največjo možnost spreminjanja prenosnih parametrov. Tudi nastavitev je zelo elegantna. S pritiskom na tipko »Mode« se nam na zaslonu prikažejo različne možnosti, ki jih izbiramo s pritiskom na tipko »Return«. Izbiramo lahko med prenosnimi hitrostmi 75, 110, 300, 600 in 1200 baudov, dolžinama podatkov 7 ali 8 bitov, pariteto 0, in 2, pomik na začetek vrste s premikom v novo vrsto ali brez njega (CR, CR+LF), dolžina kode 7 ali 8 bitov, koda tipkovnice, »external ready«, ki pošiljatelju sporoča, da je tiskalnik pripravljen za sprejem, da ali ne.

Tudi EXD 10 ima zelo lepe črke. Čeprav je matrika pik precej redkejša, »le« 16 × 10, je treba kar ostrih oči, da razliko odkrijemo. Hitrost pisanja je precej nižja kot pri EP 44, 11 znakov na sekundo. Tako kot pri EP 22 je število znakov v vrstici omejeno na 75, vsi izpisani znaki se prikazujejo na zaslonu iz tekočih kristalov.

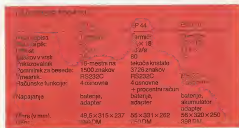
Pri tem pisalnem stroju ni mogoče spreminjati prenosnih parametrov. Hitrost prenosa je ena sama in to 600 baudov. Zato pa ima EXD 10 nekaj drugega, česar ne najdemo pri obeh Brotherjevih pisalnih strojih oziroma tiskalnikih. Ima tri stalne in pet prostih naborov znakov, ki jih lahko definiramo sami (kar nam omogoča, da si sami pripravimo slovenske č, š, ž). Izbiramo jih s tremi DIP stikali, ki pa so precej težko dostopna, tako da za nastavitev potrebujemo dolg, koničast predmet in precej smisla za ročno delo.

Pri EXD ni razlike med pisanjem in tiskanjem. Prednost je, da lahko vse osnovne nastavitve uporabimo, na primer nastavitev širine robu, tudi pri tiskalniku. Vendar pa moramo biti ravno pri nastavitvi robu previdni. Če desni rob nastavimo na primer na 40 znakov in nato izpisujemo besedilo, ki je širše, bodo vse besede odrezane na tem mestu in tiskalnik jih ne izpiše, temveč nas le opozori s piskom, da je besedilo širše od desnega robu. Zato moramo besedilo prej zožiti na potrebno mero.

EXD razume največ krmilnih kod od vseh treh omenjenih tiskalnikov. Z njimi nastavljamo levi in desni rob, določamo, brišemo in uporabljamo tabulatorje, pomikamo papir naprej in nazaj.

Oba Brotherjeva pisalna stroja sta predvsem pisalna stroja in šele nato tiskalnika. EXD 10 je bolj tiskalnik, zato tudi nima pomnilnika za besedilo.

Komu bodo ti tiskalniki prišli prav? Kdor predvsem izpisuje daljše programe, naj si omisli pravi tiskalnik. Papir oziroma trakovi zanj so cenejši in lahko uporabljajo neskončni papir. Kdor pa izpisuje predvsem besedila, pa si bo gotovo lahko pomagal z veliko kakovostjo izpisa EP 44 in EXD 10, ki prekaša navadne matrične tiskalnike.



	[illegible]	EP 44	[illegible]
...izpisa	[illegible]	...ermič...	[illegible]
...ka pik:	[illegible]	... × 18	[illegible]
...ost	[illegible]	...6 z/s	[illegible]
...nakov v vrsti	[illegible]	80	[illegible]
Prikazovalnik	16-mestni na	tekoče kristale	
Pomnilnik za besede:	1500 znakov	3726 znakov	
Vmesnik:	RS232C	RS232C	[illegible]
Računske funkcije:	4 osnovne	4 osnovne + procentni račun	[illegible]
Napajanje	baterije, adapter	baterije, adapter	baterije, akumulator adapter
Mere (v mm):	49,5×315×237	55×331×262	56×320×250
Cena:	...50 DM	750 DM	398 DM

Tudi počasi se pride do rešitve

Sinusna krivulja na C-64

Računalnik lahko služi kot odlično orodje za predstavitev različnih matematičnih funkcij. Tudi C-64 v tem ni izjema. Toda način za dosego tega predstavlja malce večji problem.

Za grafično predstavitev funkcijske vrednosti v dani točki izračunamo najprej ustrezno funkcijsko vrednost, upoštevajoč seveda še odmik in amplitudo. Kadar se nahajamo v kartezičnem koordionatnem sistemu, nam to pri enostavnih funkcijah ne predstavlja večjega problema. Težje pa postane takoj, ko računalnikov izvajalnik (interpreter) ne omogoča ekvivalentne predstavitve.

Tako je žal tudi pri C-64, kadar uporabljamo seveda že vgrajeni izvajalnik (basic V2.0).

Program SINUS prikazuje, kako se lahko lotimo takega problema. Takoj velja opozoriti, da je osnovna značilnost tega programa počasnost! Vendar pa ga je treba upoštevati predvsem kot primer ene izmed rešitev.

Poleg barvanja in brisanja ekrana, ki ga lahko lepo opazujemo je treba potem za vsak x izračunati funkcijsko vrednost (v našem primeru sin(x)). Temu pa sledi še Pretvorba funkcijske prednosti in zapis točke na ekran.

Na ekranu se bodo čez kar precej časa izrisale točke, ki bodo predstavljale grafično predstavitev sinusne krivulje od 0 do 360 stopinj.

Jani Hočevar

```
00000 REM  +------------------------------+
00001 REM  +                              +
00002 REM  +      SINUSNA KRIVULJA        +
00003 REM  +------------------------------+
00004 REM  +                              +
00005 REM  +     IZDELAL JANI 1985        +
00006 REM  +------------------------------+
00007 REM  X - POLOZAJ NA X OSI
00008 REM  Y - VREDNOST FUNKCIJE V TOCKI X
00009 REM  C - SPLOSNI STEVEC
00010 REM  +------------------------------+
00011 REM  +                              +
00012 REM  +VKLJUCITEV VISOKE LOCLJIVOSTI+
00013 REM  +                              +
00014 REM  +------------------------------+
00016 POKE 53272,PEEK (53272) OR 8:REM POSTA
      VI NASLOV EKRANA
00018 POKE 53265,PEEK (53265) OR 32:REM POST
      AVI HI-PES BIT NA 1
00020 REM  +------------------------------+
00021 REM  +                              +
00022 REM  +      BARVANJE EKRANA         +
00023 REM  +------------------------------+
00024 REM  + RUMENO OZADJE CRNE TOCKE     +
00025 REM  +                              +
00026 REM  +------------------------------+
00027 FOR C = 0TO 999
00028 :POKE 1024+C,7
00029 NEXT
00040 REM  +------------------------------+
00041 REM  +                              +
00042 REM  +      BRISANJE EKRANA         +
00043 REM  +                              +
00044 REM  +------------------------------+
00045 FOR C = 0TO 7999
00046 :POKE 8192+C,0:REM  BRISANJE (VREDNOST
      0!)
00049 NEXT
00050 REM  +------------------------------+
00051 REM  +                              +
00052 REM  +      VLECI CRTO              +
00053 REM  +                              +
00054 REM  +------------------------------+
00057 FOR C = 12036TO 12352STEP 8
00058 :POKE C,255
00059 :NEXT
00060 REM  +------------------------------+
00061 REM  +                              +
00062 REM  +      RISANJE FUNKCIJE        +
00063 REM  +                              +
00064 REM  +------------------------------+
00067 FOR X = 0TO 319:REM  X TECE PO VSEH TOC
      KAH
00068 :Y = INT (100-80*SIN (X**2/320)):REM  V
      PISI FUNKCIJO
00070 :FOR N = 0TO 24
00074 :H1 = N*8-1:H2 = (N+1)*8
00076 :N1 = 8192+N*320+8*INT (X/8)+Y-8*N
00082 :IF (Y>H1) AND (Y<H2) THEN BY = N1:N =
      24:GOTO 100
00090 :NEXT N
00100 REM  +------------------------------+
00101 REM  +                              +
00102 REM  +      VREDNOST BYTA           +
00103 REM  +                              +
00104 REM  +------------------------------+
00108 :BI = 8*(1+INT (X/8))-X-1:REM  IZRACUN
      BITA
00112 :IF PEEK (BY)<>0 THEN POKE BY,PEEK (BY)
       OR 2↑BI
00120 :IF PEEK (BY) = 0 THEN POKE BY,2↑BI
00130 NEXT X
00150 GOTO 150
```



Čudež pod pisalno mizo

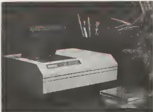
čitalcem magnetnega traku, vrstičnim tiskalnikom, operacijskim sistemom VMS ter Fortran prevajalnikom 75.000 dolarjev). Malo, če se spomnim na srečanje z nekom, ki je pred približno desetimi leti kupil računalnik PDP-1 z 8 Kb (res — osem tisoč zlogov) za 450.000 dolarjev.
Najbolje opremljeni MicroVAX pride na tržišče s 5 Mb spomina, tremi Winchester diski s po 71 Mb in kasetnim pogonom za magnetni trak s 95 Mb ter lahko spodobno oskrbuje z računalniškimi zmogljivostmi do 16 uporabnikov hkrati. Stane pa solidnih 44 000 dolarjev.

Zadrega z imeni

Vse kaže, da bomo prej ali slej prišli v zadrego z imeni za posamezne razrede računalnikov. Mikroračunalniki, miniračunalniki, superračunalniki... In sedaj še supermikroračunalniki. Nekdo bo resno moral uvesti red pri teh imenih in jasno povedati, kam kaj spada. Moj IBM-AT, na katerem pišem tale prispevek, spada med mikroračunalnike, prav tako pa tudi Sinclairova Mavrica, na kateri se igra moj sin. Stari PDP-11/34, na katerem sem včasih delal, naj bi bil miniračunalnik, čeprav se po zmogljivosti komaj more primerjati z AT. In prav tako se imenuje »mini« cela družina računalnikov, ki jih proizvaja Digital Equipment Corporation (DEC), od VAX-11/725, ki je najmanjši med njimi, do VAX 8600, zastavne ladje te družine. Čisto nemogoče je začrtati neko ločnico med posameznimi razredi. Morda po vrsti procesorja? Naj bi 8-bitni bili mikro in 16-bitni mini? Po velikosti ohišja? Takšni, ki gredo v kovček so še mikro in večji so že mini? Po številu ukazov, ki jih zmore procesor izvesti v eni sekundi? Kakršnokoli klasifikacijo bi uvedli, bi se podrla najkasneje v letu dni. In dokaz za to je najnovejši iz VAX družine, MicroVAX II, ki ga je DEC predstavil 14 maja, skupaj z nekaterimi drugimi novimi proizvodi.

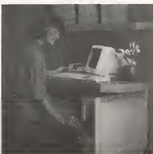
Po imenu je supermikroračunalnik — po zmogljivosti pa ga primerjajo z enim izmed večjih iz družine VAX — z VAX-11/780. Je naslednik MicroVAX I, ki je na tržišču od sredine lanskega leta dalje, vendar pa je zasnovan okoli povsem novega procesorja. Seveda je to 32-bitni procesor (kot vsi ostali VAX), vendar pa je prvi, ki je zasnovan na enem samem vezju. Naloga, ki so jo morali opraviti DEC inženirji ni bila preprosta. VAX 780 procesor ima približno 1,250 000 tranzistorjev in vso funkcionalnost tega procesorja je bilo treba spraviti na eno vezje z okoli 125 000 tranzistorji. Resnici na ljubo je treba povedati, da rešitev ni bila povsem elegantna. Skoraj 300 ukazov, ki jih VAX procesorji razumejo, so razdelili na tri skupine in ugotovili, da je 50 takšnih, za katere mikrokoda zavzema 80 % prostora v procesorju, izvajajo pa se le zelo redko. Drugih 70 ukazov je namenjeno za operacijo s plavajočo decimalno vejico (floating point instructions). In rešitev je tu: procesor v MicroVAX II, obdela 175 ukazov z lastno mikrokodo, za realna števila ima v pomoč koprocesor z dodatnimi 43 000 tranzistori in tistih 50 redkih, vendar zapletenih ukazov je obdelanih v operacijskem sistemu. Seveda se ti ukazi izvajajo mnogo počasneje, ker pa jih je v povprečju le en sam odstotek, se to ne pozna dosti (v tej skupini so PDP-11 ukazi, ki jih VAX procesorji razumejo, nadalje ukazi za obdelavo številčnih podatkov, v katerih posamezne številke zavzemajo 4 bite, ter ukazi za obdelavo teksta). Za tiste, ki poznajo te ukaze, naj povem, da sta dva od slednjih le ostala v mikrokodi, to sta MOVC3 in MOVC5.

Osnovni (in najcenejši) model ima 2 Mbyte spomina in prostora za vgraditev dodatnih sedem — do polne zmogljivosti 9 Mbytov, (naj torej razvrščamo računalnike po količini spomina?), Winchester disk z 31 Mbyti in dva disketna pogona za diskete s po 400 Kbyti. In cena? Zmogljivostim primerna, seveda. Ta model stane približno 18.000 dolarjev. Veliko, če jo primerjamo s cenami mikro računalnikov, in malo v primerjavi s cenami ostalih VAX (za primerjavo naj povem, da nas je lani stal rabljeni VAX-11/750 z 2 Mb spomina, 125 Mb diskom,



Čitalna enota za optične diske





MicroVAX II

Seveda pa ves ta spomin še ne pomeni da je tu konec zmogljivosti. Ker je tudi MicroVAX se vedno VAX (kratice pomenijo Virtual Address extension) tudi ta prevaja resnične naslove v navidezne in za le-te ima na voljo 4 Gbytov. Seveda je navidezni naslovni prostor omejen s prostorom na disku — ampak možnosti so tu in le počakati je treba da bo MicroVAX opremljen z večjimi Digitalovimi diski.

Operacijski sistem je isti kot na »pravih« VAX-VMS, kar pomeni, da bodo vsi programi, ki tečejo na njih, tekli tudi na MicroVAX — teh pa je veliko. Za tiste, ki jim je pri srcu operacijski sistem UNIX in programski jezik C je na voljo Digitalova izvedba ULTRIX32, ki je prav tako kot VMS dostopna v MicroVAX izvedbi.

Celo to majhno čudo je spravljeno v zaboj z 82 cm višine, 33 cm širine in 70 cm globine, in gre zlahka pod vsako pisalno mizo.

Pri vsej tej računalniški zmogljivosti pod mizo je treba najti kaj primernega tudi za na mizo. Morda bazo podatkov, shranjeno na 12-centimetrskem disku s 600 milijoni zlogov? To ni pomota, to je digitalni optični disk s 600 Mbyti podatkov. In cela čitalna enota ni večja od škatle za čevlje. Optični disk je majhna plastična plošča, na kateri so zapisani biti podobno kot na magnetnem disku — le da so tu enice zapisane kot mikroskopsko majhne vdolbine v površini, ki odbija svetlobo in ničle so seveda površina sama. Ta zrcalna površina je prekrita s prozorno zaščitno plastjo in disk sam ne zahteva nič bolj pazljivega ravnanja kot gramofonska plošča.

Čitalec optičnega diska uporablja za čitanje podatkov laserski žarek, ki ugotovi, kje so vdolbine in kje je površina diska nedotaknjena, in to zaporedje prevede v električne signale, tako kot običajna disketna enota. Ker lahko laserski žarek potuje po relativno dolgi poti ne da bi izgubil ostrino, je čitalna glava več kot cel milimeter nad diskom, neprimerno več kot pri magnetnih diskih. Diske lahko menjamo podobno kot gramofonske plošče ter jih prenašamo brez posebne skrbi.

Podobnost z gramofonskimi ploščami pa se še ne neha. Ti optični diski so tudi narejeni v podobnem smislu kot gramofonske plošče. Podatki se najprej zberejo na magnetnem traku in s tega traku se prepišejo na matični disk. Iz matičnega diska se nato stiskajo kopije. Tako kot pri gramofonskih ploščah se torej podatki ne zapišejo naravnost na optični disk.

Iz vsega tega sledi, da lahko optične diske le čitamo — računalnik ne more nanj zapisovati nikakršnih podatkov. In proizvodna cena tudi ni majhna, je pa odvisna od števila kopij, ki se odtisnejo s posameznega matičnega diska in postane ekonomična pri najmanj 100 odtisnjenih kopijah. Ker torej podatkov na optičnem disku ne moremo spreminjati, je ta način shranjevanja primeren za stabilne baze podatkov — ali pa za prenos velike količine podatkov. Če bi hoteli poslati preko telefonske zveze z zmogljivostjo 1200 baudov, bi taka operacija trajala 46 dni — en mesec in pol!

600 Mbytov predstavlja približno 1600 običajnih disket ali približno 200 000 tipkanih strani. Kako obsežna je že cela Britanska enciklopedija?

Sašo Novak



delov. Dobro je, če ima tiskalnik svojo mizico, tako da se tresljaji pri pisanju ne prenašajo na računalnik. Za dobro delovanje tiskalnika je najpomembnejša preventiva. Papir moramo zmeraj vlagati pravilno. Pot papirja je točno določena. iz ene škatle skozi tiskalnik v drugo škatlo in velike variacije niso priporočljive. Zobje na ozobčenih kolesih morajo papir vleči naravnost. Če se papir v tiskalniku zatakne in začne nabirati, se tiskalnik lahko poškoduje. Tiskalniki imajo sicer vgrajeno zaščito; če z roko ustavimo glavo ali valj, se tiskalnik ustavi. (To ni nasvet!) Zato je dobro, da smo z enim očesom ali ušesom zmeraj pri njem. Lahko se zgodi, da je robna perforacija poškodovana, česar pa pri polnem kartonu ne moremo videti in zato kljub sicer pravilno vstavljenemu papirju in drugo dokumentacijo za uporabo računalnika.

— Mehanski deli so najbolj občutljivi na prah.

— Elektronski deli so najbolj občutljivi na vlago in visoko temperaturo. Zato moramo zagotoviti dobro hlajenje.

— Napajalna napetost mora biti čimbolj konstantna.

Vzdrževanje zaslona in tipkovnice

S kakovostnim zaslonom in tipkovnico ne bomo imeli težav. Kdaj pa kdaj očistimo zaslon in ohišje tipkovnice. Paziti pa moramo, da tekočina ne priteče v tipkovnico, kajti pod tipkami so elektronski sestavni deli.

Vzdrževanje osebnih računalnikov

Nekaj nasvetov ne bo odveč

Računalnik povečini sestoji iz elektronskih delov, ki so zelo odporni na prah, udarce in tresljaje. Večina težav, ki jih imamo z elektronskimi deli, nastane zaradi slabega hlajenja. Ponavadi se najbolj greje napajalna enota. Če je hlajenje iz tega ali onega vzroka slabo, zaradi pregrevanja pride do premočnega raztezanja in tiskanina se tam, kjer je tak »vroč« element prispajkan nanjo, lahko zatrga. Računalnik enkrat dela, drugič pa ne. Zato moramo redno čistiti hladilne odprtine, še zlasti pogosto, če ima računalnik vgrajen ventilator in je v prostoru veliko prahu. Na računanik ne smemo postavljati predmetov, ki lahko bistveno poslabšajo hlajenje. Upoštevati moramo tudi navodila proizvajalca, ki ponavadi določi, koliko prostega prostora mora biti za računalnikom in nad njim za pasivno hlajenje.

Če računalnik neha delati, so v treh četrtinah primerov vzrok motnje v delovanju zunanjih pomnilnih enot. Občutljive so predvsem disketne enote, trdi diski so bolj odporni. Na kaj moramo paziti pri delu z disketami?

— Ne smemo jih polagati na predmete, ki ustvarjajo magnetna polja, telefon, televizijski sprejemnik, trajni magnet, vsi predmeti, ki imajo močnejše transformatorje.

— Ne smejo priti v stik s tekočinami in v prostoru mora biti čim manj prahu.

— Ko jih vzamemo iz disketne enote, jih moramo takoj dati nazaj v ovitek.

— Proste površine diskete ne smemo nikoli prijeti z roko.

— Po disketi ne smemo pisati ali jo pripenjati s pisamiškimi sponkami.

— Na diskete ne smemo odlagati knjig ali drugih predmetov.

— Disket ne smemo pregibati.

— Mesto, kjer shranjujemo diskete, ne sme biti izpostavljeno sončnim žarkom. Idealna temperatura za shranjevanje je med 10° do 50 °C pri relativni vlagi med 8 in 80 %. Pozor: v zaprtem avtomobilu se temperatura poleti lahko hitro dvigne čez kritično mejo.

Če disketo že dalj časa uporabljamo in so na njej zaradi obrabe že močno vidni krogi, lahko pričakujemo težave pri branju. Če na primer obdelujemo besedilo, se nam lahko zgodi, da bo računalnik del besedila »spustil«, ker ne najde bloka podatkov ali pa bo na zaslon izpisal kakšen mišmaš.

Največ dela je s tiskalnikom, ki ima največ premičnih mehanskih namesto težko pričakovanih rezultatov dobimo papirno solato. Vsake pol leta ali vsakih 500 delovnih ur naj bi strokovnjak tiskalnik očistil, naoljil ležaje in pregledal mehaniko. Ker je pri nas takšnih strokovnjakov toliko kot nič, je najbolje, da kar sami postanemo strokovnjak za čiščenje. Večina bolj solidnih proizvajalcev priporoča in prodaja posebna maziva za svoje tiskalnike. Vsaj enkrat na teden pa moramo odstraniti prah in papirne ostanke.

Osnovna dejstva in pravila za dobro vzdrževanje

— Na splošno velja: čim več mehanskih in gibljivih delov ima neka naprava, tem več je možnosti za okvare.

— Dobro preberimo priročnike

Vzdrževanje disketne enote

Enkrat letno očistimo pisalno bralno glavo s priporočenim čistilnim sredstvom za disketne enote.

Trdi disk

V enakomernih časovnih razmakih (odvisno od uporabe) očistimo plošče s posebno krpo in posebnim čistilnim strojem.

Vsake pol leta zamenjamo filter za zrak.

Enkrat letno očistimo pisalno bralne glave s posebnim čistilnim sredstvom (kupimo v ustrezno opremljeni trgovini)

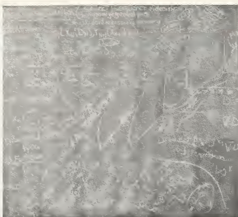
Tiskalnik

Enkrat tedensko odstranimo ostanke papirja in prah.

Na vsakih 500 ur delovanja podrobno pregledamo, očistimo in namažemo mehanske dele.

ALEŠ BURGAR

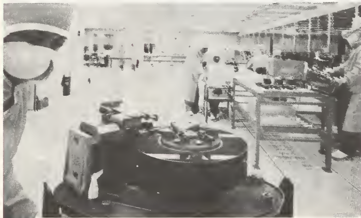
Šest velikih evropskih proizvajalcev, francoski Bull, nemška Siemens in Nixdorf, nizozemski Phillips, angleški ICL in italijanski Olivetti je sklenilo pakt imenovan »Open group for Unix system«, odprta skupina za sistem unix, kar pomeni, da se jim lahko pridruži vsak evropski proizvajalec računalnikov.
Tu ne gre za običajni trgovinski dogovor, temveč očitno za novo potezo v od dne do dne ostrejši vojni z IBM na rastočem računalniškem tržišču.
Sedanja tehnološka vojna močno spominja na spopad med sistemoma Pal in Secam v šestdesetih letih, ko je šlo za to, kdo bo dobil primat na nastajajočem trgu barvnih televizijskih sprejemnikov. Vendar pa je bil tedanji spopad v primerjavi s planetarnimi razsežnostmi bitke za standardne operacijske sisteme v informatiki prava igrača.



»Mehka« vojna Evropa: IBM

Kako se upreti premoči velikana?

Kaj so takoimenovani standardni operacijski sistemi? Gre za sistemske programe, ki omogočajo prenosljivost uporabniških programov med sicer različnimi računalniki. Operacijski sistem lahko primerjamo z živčevjem v človeškem telesu. Opravlja ogromno število operacij, ne da bi se tega sploh zavedali. Tako kot naš živčni sistem prilagaja dihanje glede na količino kisika v zraku in potrebe, sporoča ukaze mišicam in nadzoruje bitje srca, tako operacijski sistem računalnika opravlja vse krmilne funkcije, ki so očem uporabnika sicer skrite, vendar bistvene za delovanje. Operacijski sistem je tudi »okolje«, v katerem delajo uporabniški programi. Knjigovodski program ali urejevalnik nima neposrednega stika s strojem, temveč z njim komunicira prek operacijskega sistema, torej programa, ki deluje kot posrednik med strojem in uporabniškim programom.



Zato morajo biti ti programi napisani, tako da se skladajo z operacijskim sistemom, tu pa smo že pri dejstvu, ki igra odločilno vlogo pri nakupu računalnika. Še pred desetimi leti so podjetja kupovala opremo za elektronsko obdelavo podatkov na osnovi karakteristik strojne opreme, zavedajoč se, da so se z izbiro proizvajalca povezali ne le z določeno tehnologijo, temveč si predvsem odprli pot do določene skupine programov.

Danes je vse čisto drugače: na področju mikroinformatike so se pojavili določeni standardi, tako da potencialni kupci pri izbiri ne upoštevajo le zmogljivosti stroja, temveč predvsem razpoložljivi uporabniški program za izbrani operacijski sistem. Čeprav se je IBM za poseg na tržišče osebnih računalnikov odločil razmeroma pozno, je napravil dve dobro premišljeni potezi: izdelal je računalnik na osnovi mikroprocesorja drugega proizvajalca, ga torej ni razvil v svojih laboratorijih, temveč ga je kupil pri Intelu, za operacijski sistem pa se je obrnil na neodvisnega proizvajalca, na Microsoft. Dva bistvena dela računalnika sta bila torej znana in vsem dostopna. Vsakdo si je v trgovini za vogalom (to velja za ZDA) kupil isti čip in uporabljal isti operacijski sistem, ki ni bil zaščiten s patentom IBM (razen majhnega dela). Posrečena izbira. Tako je lahko vsakdo začel izdelovati programe za PC. Res je, da je IBM izgubil celovito kontrolo, ker programe izdelujejo in prodajajo neodvisne družbe, po drugi plati pa je od tega vendarle imel korist. Krog programov za PC se je bolj in bolj širil, računalnik je bil vse bolj iskan in vse bolj standarden. Tako si lahko razložimo pravi razcvet modelov »združljivih z IBM« in drugih podobnežev, ki uporabljajo iste uporabniške programe pri ponavadi nižji ceni in kakšnem kolescu več. Temu standardu so se morala ukloniti vsa prestižna podjetja: Olivetti-ATT, Ericson, Sperry, Wang, Digital in tudi Commodore, Bull-Honeywel in Hewlett Packard.

Glavni razlog za to, da vsi lovijo IBM, je prav programska ali mehka oprema. Kupec danes hoče vedeti predvsem to, kaj bo lahko konkretno delal s kupljenimi računalniki. To pa je v največji meri odvisno od količine napisanih programov ali od tega, ko-



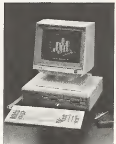
liko programov je v pripravi. Količina programov pa je odvisna od tega, koliko se je standard nekega računalnika uveljavil na tržišču. V ZDA je danes najmanj 30.000 majhnih programskih družb, ki proizvajajo uporabiške programe za MS-DOS, operacijski sistem, ki ga je uveljavil IBM PC. Proizvajalci osebnih računalnikov upajo, da bodo do leta 1990 podvojili promet predvsem na osnovi proizvodnje zmeraj novih programov. Leta 1984 so na svetu prodali 3 milijone 200 tisoč računalnikov, za leto 1989 pa International Data Corporation računa na 10 milijonov in 200 tisoč kosov. Po analizi družbe za tržne raziskave iz Massachusettsa, Forrester Research, ki jo povzemamo iz ameriške revije »High Techology«, bo IBM pri nadaljevanju sedanje rasti do takrat osvojil 60—70 % tega ogromnega tržišča. Ta predvidevanja potrjujejo tudi zadnje izjave novega predsednika IBM Johna F. Akersa. IBM, ki ima že danes največji dobiček med vsemi velikimi družbami, bo do leta 1990 početveril promet. Takrat bo mehka oprema namesto z današnjimi 10 % v njem zastopana s 30 %.

Pred temi hegemonističnimi grožnjami so evropski proizvajalci ne le v nevarnosti, da bodo igrali vse manj in manj pomembno vlogo, temveč da bodo zlepa ali zgrda izginili. Na tržišču, ki se tako naglo razvija, se ni mogoče skriti v luknjo in »potuhniti« v dosedanjih standardih. Do srede 70. let, ko so osebni računalniki imeli 8-bitne mikroprocesorje (obdelovali naenkrat številčno zaporedje osmih enic ali ničel), sta bila standardna operacijska sistema Appla in Digital Researcha in zanju je bilo na trgu obilo programov. Ko pa so se pojavili 16-bitni mikroračunalniki in je prevladal operacijski sistem MS-DOS, ki ga je »vsilil« IBM, sta s tem izgubili velik del tržišča.

Nekaj podobnega bi se utegnilo primeriti ob prehodu s 16-bitnih na 32-bitne mikroprocesorje. Po tej metamorfozi se bodo na mizah pojavili resnično zmogljivi in hitri osebni računalniki. Takrat bomo morda že težko govorili o osebnem računalniku, bolj ustrezen izraz bo »supermikro«, če pa bomo prevzeli izraz od anglofonih ljudstev, pa jih bomo imenovali AWS, »razvite delovne postaje«.

Kdo bo torej dobil veliko vojno za standard, kateri smo priča? Dogovor med še stotimi evropskimi proizvajalci je povezan z vstopom svetovnega telekomunikacijskega velikana AT&T v informatiko. AT&T, ki bo v prihodnjih letih edini pravi nasprotnik IBM, ima lasten standarden operacijski sistem, ki se imenuje unix in ki so ga razvili v Bellovih laboratorijih, v najpomembnejšem raziskovalnem središču na svetu. Ustvaril ga je leta 1969 programer Ken Thompson, ki se je pri delu zmeraj znova jezil nad nedoslednostmi in omejitvami tedaj obstoječih operacijskih sistemov. Unix je v resnici idealen za prenašanje z enega računalnika na drugega. Leta in leta so ga izboljševali na različnih svetovnih univerzah. Inačico, ki je nastala v Berkeleyu, so skoraj zastonj pošiljali drugim univerzam; zadoščala je pismena prošnja. AT&T je bil s tem zadovoljen, ker je v miru monopolno vladal telefonskemu omrežju in podarjal denar in Nobelove nagrade za »namenske« raziskave. Tako se je sistem unix počasi uveljavil v računalniškem svetu.

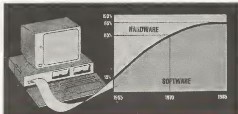
To je ozadje dogovora šesterice za standardizacijo unixa. Skupina Unix si prizadeva, da bi zagotovila skladnost z drugimi mednarodnimi standardi (na primer z ISO, International Standard Organisation). Angleški ICL se je zganil prvi, drugi so mu hitro sledili. Skupina obljublja, da bo zares odprta, da bo torej objavila in omogočila dostop do standardov, ki jih bo začela razvijati že v letošnjem letu na področju pro-



gramskih jezikov, elektronski obdelavi podatkov in komunikacijah med posameznimi delovnimi postajami. To je torej povabilo programskim družbam, naj svoj vagon programov priključijo na uniksov vlak.

Ne gre za velikodušnost, temveč za dobro premišljeno in hkrati obvezujočo ponudbo. Proizvodnja kakovostne in zanesljive mehke opreme je draga stvar. Zato je bolje na stežaj odpreti vrata programskim družbam, naj delajo brez omejitev, saj s tem povečuje možnosti računalnikom, ki imajo operacijski sistem unix. Kar se tiče mednarodne konkurence pa je to tudi poskus zaustaviti IBM na evropskih obalah, preden bo prepozno.

Bo Evropa prišla iz dežja pod kap? Morda, če gledamo dolgoročno. Trenutno pa ni skrbi, saj je ameriški kolos močno zaposlen z reorganizacijo. Res je, da je lastnik uniksa, vendar pa sam ne more naprej, še najmanj pa v Evropi. Evropski proizvajalci se torej naslonijo na manj napadalnega velikana in z njegovo pomočjo ustavijo IBM. Ta pa zdaj prav gotovo pripravlja protiudarce in išče nove poti. Kaj lahko se zgodi, da se bo modri velikan tudi tokrat izkazal za presenetljivo gibčnega, in morda, če se bo unix izkazal, celo sam sprejel ta sistem za svoje računalnike.



Kako so razdeljeni stroški velikega informacijskega sistema med trdo opremo (računalnik, zunanji pomnilnik, tiskalniki, risalniki) in mehko opremo (sistemski in uporabniški programi) nam prikazuje znana krivulja imenovana 80 — 20.

V 50. letih so stroški strojne opreme predstavljali 80 % skupnih nabavnih stroškov. Kasneje pa je vrednost mehke opreme rasla hitreje, tako da je danes v skupnih stroških zastopana z 80 %. Cena trde opreme je ves čas bolj ali manj konstantna, mehka oprema pa je vse bolj zapletena in draga. Dimenzije računalnikov se zmanjšujejo, njihova zmogljivost se povečuje, zahtevajo nove in nove programe. Po drugi strani pa je v mehko opremo vloženega toliko denarja, da je ni mogoče kar tako zavreči in zato včasih mehka oprema zavira razvoj trde, hkrati pa prodiranje računalništva na nova področja zahteva nove programe za nove uporabnike: programe, ki morajo biti lahki za uporabo, razumljivi in prijazni.

ukvarjal s programiranjem, sem bil prepričan, da sem najboljši programer na svetu. Znal sem uporabljati pet različnih programskih jezikov, znal sem napisati 3000 vrstic dolg program, ki je resnično DELAL. Nato pa sem zašel v Resnični Svet in dobil nalogo da preberem in razumem 200.000 vrst dolg program in mu zmanjšam čas izvajanja za polovico. Vsak Pravi Programer bo znal povedati da vse »strukturirano programiranje« na

konec koncev za igranje in celo stare dame in osnovnošolci ga lahko razumejo.

Unix je seveda dosti bolj zapleten — povprečni uporabnik Unixa niti ne ve, kako se ukaz PRINT imenuje ta teden. Vendar kljub temu Unix ni kaj dosti več kot poveličana video igra. Nihče ne more opraviti Resnega Dela na Unix sistemu — kvečjemu pošilja šale naokoli, piše pustolovske igre ali znanstvena dela.

Pravi programer ne nosi kravate

Naj živi Fortran!

V dobrih starih časih — v »zlati dobi« računalnikov — je bilo lahko ločiti može od otrok (v literaturi včasih imenujejo prve tudi »pravi možje« druge pa »polentarji«). Takrat so bili Pravi Možje tisti, ki so razumeli računalniško programiranje, tisti pa, ki ga niso, so bili »polentarji«. Pravi računalniški programer je govoril v stavkih kot npr. »DO 10 I = 1,10« in »ABEND«, in ostali svet je odgovarjal »računalniki so preveč zapleteni zame« in »ne trpim računalnikov — so tako neosebni«. Pravi Možje se ne bojijo biti neosebni.

Toda kot običajno se časi spremenjajo. Danes živimo v svetu v katerem male stare dame upravljajo računalnike v svojih mikrovalovnih pečicah in dvanajstletni otroci kot za šalo premagujejo Prave Može pri igranju Packmana ter si vsak lahko kupi in celo razume svoj mikroračunalnik. Pravi Programer je v nevarnosti da postane ogrožena vrsta in da ga bodo zamenjali srednješolci s svojimi Spectrumi.

Jasno je, da je treba pokazati kje so razlike med povprečnim srednješolskim igralcem Packmana in Pravim Programerjem. Če bo razlika jasna, bo dala tem otrokom nekaj, k čemur bodo lahko stremeli, vzor, očetovsko podobo. Na drugi strani pa bo postalo jasno, da Pravih Programerjev igralci Packmana ne morejo nadomestiti.

Programski jeziki

Najlažje je ločiti Pravega Programerja od ostale množice po programskem jeziku, ki ga uporablja. »Polentarji« uporabljajo Pascal. Niclausa Wirtha, ki je zasnoval ta jezik, so nekoč vprašali kako izgovarja svoj priimek. »Lahko me kličete po naslovu — VIRT, lahko pa tudi po vrednosti — VORT« je odgovoril. Iz tega odgovora je takoj razvidno, da je Nicklaus Wirth »polentar«. Edini mehanizem za posredovanje parametrov v podprogram, ki ga Pravi Programer upošteva je prenos po vrednosti, tako kot to dela IBM/370 FORTRAN-G prevajalnik. Pravi Programer ne rabi vseh teh abstraktnih konceptov, da opravi svoje delo; prav zadovoljen je s svojo tipkovnico, FORTRAN IV prevajalnikom in pivom.

— Pravi Programer piše programe za umetno inteligenco v Fortranu

— Pravi Programer obdeluje liste v Fortranu

— Pravi Programer piše knjigovodske programe (če jih sploh) v Fortranu

Če nečesa ne moreš narediti v Fortranu, naredi v zbirnem jeziku. Če ne moreš narediti v zbirnem jeziku, potem tega sploh ni vredno delati.

Strukturirano programiranje

Akademiki v računalniški znanosti so zašli na pot »strukturiranega programiranja« v zadnjih nekaj letih. Pravijo da je programe veliko lažje razumeti, če programer uporablja nekatere posebne programske metode in konstrukcije. Seveda se ne morejo zediniti katere, in primeri s katerimi hočejo dokazati svoj prav zavzamejo kvečjemu eno stran v kakšnem zakotnem znanstvenem listu — kar gotovo ni primer, ki bi prepričal kogarkoli. Potem, ko sem se že nekaj časa svetu ne bo pomagalo rešiti problema kot je ta — tu pride do izraza le pravi talent.

— Pravi Programer se ne boji uporabljati GOTO stavka

— Pravi Programer zna napisati pet strani dolgo zanko ne da bi se zmedel

— Pravi Programer ne piše komentarjev — namen prorama je razviden iz kode

— Pravi Programer piše kodo, ki se sama modificira, še posebej če s tem pridobi vsaj 20 nanosekund znotraj tesni zanki

— Ker FORTRAN IV nima IF . . . ELSE, REPEAT . . . UNTIL ali CASE stavkov Pravi Programer ni zaskrbljen, ker jih ne uporablja. Razen tega pa jih je prav lahko simulirati z GOTO stavki.

Podatkovne strukture so naslednja pomembna postavka »strukturiranega programiranja«. Abstraktni podatki, strukture, kazalci, liste itd. so postali popularni v nekaterih krogih. Vendar, kot ve vsak Pravi Programer, je edina uporabna struktura podatkov niz. Vse ostalo so samo posebne izvedbe nizov in se jih lahko kot take tudi uporablja brez da bi mešal v program nujno potrebne deklaracije. Pravi Programski Jeziki, kot vsi vemo, razpoznajo tip podatka po prvi črki imena spremenljivke (ki ima lahko le šest črk).

Operacijski sistemi

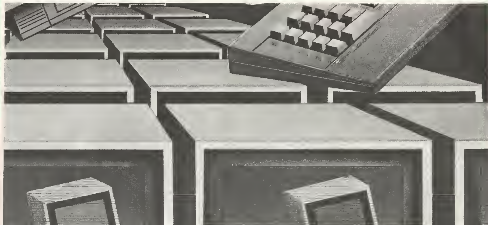
Kakšen operacijski sistem uporablja Pravi Programer? CP/M? Niti kakor ne, ta operacijski sistem je

Ne, Pravi Programer dela na operacijskem sistemu OS/370. Dober programer bo našel in razumel opis napake IJK350I, ki jo je pravkar opazil, v svojem JCL (Job Control Language) priročniku. Izredno dober programer lahko piše programe v JCL brez priročnika, in če je res izvrsten, bo našel napako v listingu šest megabytnega spomina brez da bi uporabil šestnajstiški kalkulator.

OS je res pravi operacijski sistem. Prav lahko je uničiti dneve in dneve dela z eno samo črko, ki ni na pravem mestu, zato je pozornost na višku. Nekateri ljudje sicer trdijo da je nekje <???TIMESHARING???>sistem, ki dela na OS/370, toda po daljšem študiju sem prišel do zaključka, da se motijo.

Programerski pripomočki

Kakšne vrste pripomočkov uporablja Pravi Programer pri svojem delu? Pravi Programer bi lahko spravljal svoje programe v računalnik tako, da bi jih odtipkal (v šestnajstiškem številskem sistemu, seveda) na konzolni plošči računalnika. V dneh, ko so računalniki še res imeli take plošče, se je ta način tudi resnično uporabljal. Povprečni Pravi Programer je poznal zagonsko kodo (BOOTSRAP LOADER) na pamet in jo je odtipkal naravnost v spomin vsakič, ko mu jo je njegov lastni program izbrisal. Takrat je tudi spomin bil pravi spomin — ni se izbrisal ko je zmanjkalo električnega toka. Dan-



danes spomin ali pozabi stvari, ki jih ne bi smel, ali pa si jih zapomni potem, ko so bile že dolgo pozabljene. Legenda pravi, da je Seymore Cray, oče superračunalnika z istim imenom in večine računalnikov Control Data Corporation, dejansko odtipkal prvi operacijski sistem za CDC 7600 na konzolni plošči, ko so ga prvič pognali. Ni potrebno reči, da je Seymore Pravi Programer.

Naokoli gre tudi zgodba o programerju v Texas Instruments. Lepega dne je dobil telefonski klic z drugega konca kontinenta. Klical ga je neki uporabnik, kateremu se je zrušil operacijski sistem sredi pomembnega dela. Jim je usposobil računalnik tako, da je temu uporabniku po telefonu sporočal kodo za disk I/O instrukcije in sistemske tabele, ta pa jih je odtipkoval na konzolni plošči in Jimu čital vsebine registrov.

Dandanes tudi Pravi Programerji že uporabljajo urejevalnike teksta za pisanje programov. Pri veliki izbiri teh urejevalnikov bo seveda tisti, ki ga bo Pravi Programer uporabljal, odražal njegov značaj. Veliko ljudi ima za najboljše urejevalnike nekatere od tistih, ki so bili napisani v Xerox Palo Alto Research Centru v Kaliforniji, za uporabo na njihovih Dorado in Alto računalnikih. Na nesrečo pa Pravi Programer ne bo nikoli uporabljal računalnika z operacijskim sistemom, ki se imenuje SmallTalk (obrekovanje) in ne bo nikoli uporabljal miške.

Pravi Programer meni, da je koncept »kar vidiš to tudi dobiš« prav tako neprimeren pri urejevalnikih teksta kot pri ženskah. Ne, Pravi Programer hoče »prosil si, zdaj pa imaš« urejevalnik — zapleten, neoproščajoč, nevaren vendar pa vsemogočen — TECO z eno besedo. Pravijo, da je ukaz v TECO urejevalniku še najbolj podoben znakom, ki jih dobiš na terminalu pri moteni liniji. Ena od zabavnih iger v TECO je ta, da odtipkaš svoje ime kot ukaz in poizkušaš uganiti kaj se bo zgodilo. Skoraj vsaka tipkovna napaka bo verjetno uničila program — ali še slabše, vrinila skrivnostno kodo, ki se bo pojavila kot napaka nekje v programu, ki je nekoč delal. Zato se Pravi Programer prav nerad dotakne programa, ki že skoraj dela, z urejevalnikom teksta. Veliko lažje je vriniti popravek naravnost v binarno strojno kodo s čudovitim programom SUPERZAP na IBM računalnikih ali enakovrednim na ostalih. Ta način je tako uspešen, da veliko programov, ki delajo na IBM računalnikih nima več nobene povezave z originalno kodo v Fortranu. V veliko primerih originalne kode niti ni več nikjer najti. Kadar je treba popraviti napako v takem programu, ne bo nihče poslal kaj manj kot Pravega Programerja na delo — noben »polentar« niti ne bi vedel kje začeti.

— Pravi Programer ne uporablja sistemov za vzdrževanje kode. Svojo ima zaklenjeno v nekem predalu — ker je tako pomembna

<???DEBUGER???> za izvirni programski jezik — ker zna čitati listing spomina



— Pravi Programer ne uporablja prevajalnikov, ki preverjajo meje nizov. Le ti izničijo zanimivo uporabo EQUIVALENCE stavkov in prečijo spreminjanje sistemskih programov z uporabo negativnih <???SUBSCRIPTov???>.

Pravi programer pri delu

Kje povprečni Pravi Programer dela? Kakšni programi so vredni pozornosti in truda nekoga, ki je tako idividualen in talentiran? Prepričani ste lahko, da niti mrtvega ne boste našli pisati knjigovodskih programov v Cobolu ali programe za sortiranje naročniških seznamov za časopise.

— Pravi Programer dela v Los Alamos National Laboratory in piše programe za simuliranje atomskih eksplozij na Cray superračunalniku

— Pravi Programer dela za Boeing in snuje operacijske sisteme za raketne izstrelke

— zasluga mnogih Pravih Programerjev pri NASA je bila, da je človek stopil na Mesec

Nekaj najboljših Pravih Programerjev dela v Jet Propulsion Laborotories v Kaliforniji. Ti poznajo na pamet operacijske sisteme vesoljskih ladij Pioneer in Voyager. S kombinacijo obsežnih fortranskih programov na Zemlji in majhnih asemblerskih programov v vesoljski ladji so sposobni napraviti čudeže — zadeti deset kilometrski krog na Saturnu po šestih letih potovanja skozi vesolje.

Vesoljska ladja Galileo bo uporabila marsovo gravitacijo in trajekcija jo bo vodila 80 kilometrov (toleranca je 3 kilometre) nad površino tega planeta. Nihče pri zdravi

Naj živi Fortran!



pameti ne bo zaupal Pascalu (ali programerju, ki piše v Pascalu) navigacije pri takšnih tolerancah.

Kot vidite, precej Pravih Programerjev dela za ameriško vlado, pravzaprav za obrambno ministrstvo. In prav tu so se pred časom začeli zbirati temni oblaki na njihovem obzorju. Neka velika živina v obrambnem ministrstvu se je namreč odločila da morajo biti vsi programi napisani v programskem jeziku ADA. Nekaj časa se je zdelo, da je ta jezik napisan v pravem nasprotju z vsemi načeli Pravega Programiranja — z vsemi podatkovnimi strukturami, tipi podatkov in podpičji! Na srečo pa ima dovolj zanimivih lastnosti, ki jih bo Pravi Programer znal uporabiti. Je neverjetno zapleten, pozna načine za vpletanje v operacijski sistem in preurejanje spomina in — Edgar Djikstra ga ne mara. Kot morda še ne veste, Djikstra je avtor dela »GOTO stavki so škodljivi«, mejnika v metodologiji programiranja, ki mu ploskajo vsi, ki programirajo v Pascalu in podobni. Razen tega pa — Pravi Programer lahko piše fortranske programe v katerem koli programskem jeziku.

Včasih bo Pravi Programer nekoliko zatajil svoje principe in se bo ukvarjal z nečem, kar bo za malenkost bolj običajno kot so atomske eksplozije, če bo le pri tem dovolj denarja. Nekaj Pravih Programerjev piše video igre za Atari, na primer.

Pravi programer pri igri

Na splošno se Pravi Programer igra prav tako kot dela — z računalniki. Neprestano je začuden da dejansko dobiva plačo za nekaj, kar bi sicer tako in tako počel za čisto zabavo (seveda je previden in tega nikoli ne pove na glas). Včasih stopi iz svoje pisarne po vdih svežega zraka in na pivo ali dve.

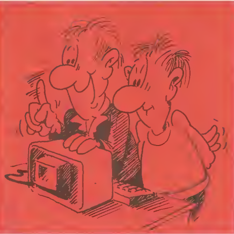
— Na zabavi lahko prepoznate Pravega Programerja po tem, da bo stal v kotu sobe in govoril o varnosti v operacijskih sistemih in kako zaobiti varnostne ukrepe,
— na plaži bo risal v pesku logične diagrame poteka programa,
— na pogrebu bo rekel »Ubogi Marko, pa mu je program za sortiranje že skoraj delal«.

Naravno okolje pravega programerja

V kakšnem okolju Pravi Programer najbolje deluje? To je pomembna postavka za vsakega šefa Pravega Programerja. Upoštevajoč ogromne zneske, ki mu jih plačujejo zato, da bi ostal pri podjetju, mu je treba zagotoviti okolje, kjer lahko opravlja svoje delo.

Tipični Pravi Programer živi pred zaslonom računalniškega terminala. Okoli tega terminala so:
— Listingi vseh programov, na katerih je Pravi Programer kdajkoli delal, v približnem kronološkem zaporedju in na vseh razpoložljivih vodoravnih površinah pisarne,
— okoli šest polpraznih kozarcev s kavo, v nekaterih plavajo cigaretni ogorki,
— razen če je izredno dober, bo imel OS JCL priročnik in Principles of Operation odprte na neki posebno zanimivi strani,
— na zidu je prilepljen printerski zapis Snoopijevega koledarja za leto 1980.

Pravi Programer je sposoben nepretrgoma delati 30, 40 ali 50 ur če je v časovni stiski. Pravzaprav mu je ta način najljubši. Slab odzivni čas na računaniku ga ne moti, še pomaga mu ujeti nekaj spanja, kadar mu prevajalnik prevaja najnovejšo verzijo programa. Če delovni načrt dopušča bolj sproščeno delo, si bo Pravi Programer poiskal spodbudo tako, da se bo prvih devet tednov ukvarjal z majhnim, vendar zanimivim delom programa, ostalo pa bo dokončal v dveh ali treh 50 urnih maratonih. Ne le da ta način spravlja njegovega šefa na rob obupa, ampak je tudi dober izgovor zakaj ni napisane dokumentacije o programu. Na splošno:
— noben Pravi Programer ne dela od devetih do petih (razen če to niso nočne ure)
— Pravi Programer ne nosi kravate
— Pravi Programer ne nosi čevljev z visoko peto
— Pravi Programer prihaja v službo ob času za kosilo
— Pravi Programer se spominja (ali pa tudi ne) imena svoje žene. Na vsak način pa seveda zna na pamet ASCII in EBCDIC tabelo
— Pravi Programer ne zna kuhati. Trgovine seveda niso odprte ob treh zjutraj, ko običajno hodi domov.



In prihodnost?

Pravi Programerji so nekoliko zaskrbljeni, ker nove generacije programerjev, ki prihajajo iz šol niso nikoli videle konzolne plošče računanika. Le redkokdo od teh še zna računati v šestnajstiškem sistemu brez uporabe kalkulatorja in vsi uporabljajo tekstovne urejevalnike, ki jim štejejo oklepaje ter »prijazne« operacijske sisteme. Najslabše od vsega je, da nekateri od teh »računalniških strokovnjakov« dobijo diplome ne da bi se naučili Fortran!

Vendar pa prihodnost ni tako črna. Niti OS operacijski sistem, niti Fortran ne kažejo znakov odmiranja, kljub vsem naporom Pascalskih programerjev po vsem svetu. In kljub temu da so se pojavile verzije fortranskih prevajalnikov, ki poznajo IF... ELSE in DO... WHILE stavke (Fortran-77) ima vsaka od njih možnost, da se spremeni nazaj v Fortran-66 prevajalnik ob enem samem ukazu in prevaja DO zanke, tako kot so bile zasnovane na samem začetku.

Celo Unix ni tako slab, kot je bil v začetku. Pojavili so se znaki, vredni pozornosti Pravega Programerja, kot na primer navidezni spomin. In če pustimo ob strani dejstvo, da je »strukturirano«, je lahko celo programiranje v C programskem jeziku izziv za Pravega Programerja. Imena spremenljivk so lahko dolga sedem (deset? osem?) črk in če dodamo intelektualne možnosti, ki jih ima uporaba kazalcev je skoraj kot da bi imeli Fortran in Assembler v enem jeziku (da ne omenjam nekaterih izvimih uporab ‡DEFINE stavka).

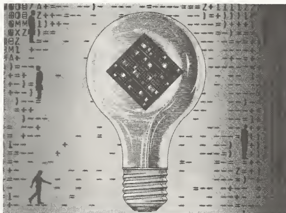
Ne, prihodnost sploh ni tako slaba. Dokler bodo obstajali slabo zastavljeni cilji, skrivnostne napake v programih in nesprejemljivi časovni roki, se bo našel Pravi Programer, ki bo rešil Problem ter prihranil dokumentacijo za kasneje. Naj živi Fortran!

Sašo Novak

Elektronska obdelava podatkov za začetnike (I)

Pametni stroji ušli iz rok posvečenih

Kako pravzaprav z računalniki, temi čudesi današnjega časa obdelujemo podatke? S katerim čarobnim izrekom si odpremo s sedmero pečati zaklenjena vrata računalniškega sveta? Je računalnik res bavbav? Množica vprašanj se poraja ob začetkih druge industrijske ali računalniške revolucije, ko »pametni« stroji uhajajo iz rok posvečenih in se selijo med ljudstvo. A računalnik je v bistvu zelo preprosta reč. Zato smo prvi članek iz serije, s katero želimo na čimbolj poljuden način predstaviti, kaj je elektronska obdelava podatkov, imenovali RAČUNALNIKI IN SODOBNA AVTOMATIZIRANA KUHINJA.



Računalnik je stroj, ki opravlja procese. Stroj, kot drugi stroji, čeprav malo drugačen. Ko govorimo o opravljanju procesov, seveda nimamo v mislih sodnih procesov, temveč dejavnosti, ki jih, pa naj se sliši še tako neverjetno, prav lahko primerjamo s tako vsakdanjo stvarjo, kot je pomivanje posode. Pustimo za sedaj vnemar dejstvo, da računalnik ne odpira pipe s toplo in mrzlo vodo in da sestoji iz centralne enote in pomnilnika in zaslona in tipkovnice in tiskalnika in kdo ve česa še.

S pomivanjem posode si je bolj ali manj vsakdo domač. **vhodni podatki**, so umazana posoda, detergent, voda in krpa ali ščetka. Izvajalec procesov — **procesor** — je človek. Kaj moramo storiti, to vemo, program imamo torej že v glavi. Rezultat procesa, **izhodni podatki** je čista, mokra posoda.

Če prvemu procesu pridružimo še drugega, namreč sušenje posode, imamo opravka s kompleksnim procesom: vhodni podatki za drugi proces so izhodni podatki prvega procesa, namreč mokra posoda, ki ji dodamo še krpo za brisanje.

Čeprav je naloga na videz zelo preprosta, pa pri delu lahko kljub temu pride do težav takrat, ko je odcejalna miza polna. Tedaj izbrana delovna strategija (najprej pomivanje, potem brisanje) odpove in oba procesa moramo opravljati izmenoma. Gospodinja ali gospodar storita to povsem samodejno in jima še na misel ne pride, da bi na odcejalnik postavila več posode, kot jo gre nanj. Pri računalniku je to seveda malo drugače: če pri podobnih nalogah zamenjave načina dela ne sprogramiramo vnaprej, računalnik neutrudno dela, dokler ni pomnilnik (odcejalnik) zvrhano poln. Tedaj se program ustavi. Bolj imenitni računalniki nam to tudi povedo, na primer z besedami »Memory overflow« (poln pomnilnik) ali pa kar na kratko »Fatal error« (usodna napaka).

Časi se spreminjajo in mi z njimi, drage bralke in bralci. Danes marsikatera žena moža čez prav rada priskoči na pomoč, ko se ubada s kuhinjskimi opravili, skratka, dela se radostno lotita timsko; pravimo da imamo **večprocesorski** ali s tujko multiprocesorski sistem. Posodo tedaj pomivamo in brišemo hkrati, delo gre hitreje in bolj učinkovito od rok in kot bi mignil, že je vse lepo za nami.

Iz vsakdanjega življenja nam je tudi znano, da lahko proces pretrgamo, če je treba: ko v sobi zaveka dojenček, z veseljem odložimo kuhinjska opravila. Tudi pri računalniku ni nič drugače. Če ima računalnik en sam procesor, ta prekine delo in z njim nadaljuje kasneje. Pri večprocesorskem sistemu pa prekinitev ustavi le en procesor, drugi pa veselo delajo naprej in morda prevzamejo še delo svojega zaposlenega kolega, če imajo kaj prostih zmogljivosti.

Naprave za elektronsko obdelavo podatkov uporabljamo za elektronsko obdelavo podatkov. Po tej kratki in jedrnati ugotovitvi, s katero pa smo zadeli žebljico na glavico, se seveda moramo vprašati, kaj so pravzaprav podatki? **Podatki** v računalniku so znaki ali kombinacije znakov. Kombinacija znakov, na primer 324314, pa dobi svoj pomen šele tedaj, ko vemo, kaj pomeni, na primer telefonsko številko. Podatki, ki jih obdeluje računalnik, morajo ustrezati določenim pravilom. Pri vsaki nalogi imamo seznam dovoljenih **podatkovnih tipov**. Pri računanju zahteva stroj na primer številke ali pa navadne črke. Če vstavimo podatke, ki ne ustrezajo izbranemu tipu — na primer ločila — potem se upre z vsemi štirimi.

Prej smo računalnikov način dela primerjali s pomivanjem posode. Pa stopimo spet v kuhinjo in naredimo tak seznam dovoljenih »podatkov« med drugim recimo določimo, da noži ne smejo biti daljši od 76 cm, lonci pa ne večji od 33 litrov, pri čemer morajo biti pripadajoče pokrovke obvezno živo rdeče barve.

Prava »skrivnost« elektronske obdelave podatkov je, kako iz pravih vhodnih podatkov dobiti prave izhodne podatke. Zato poleg programa, s katerim krmilimo potek dela, potrebujemo tudi različne pomnilnike za podatke. Pri vsakem procesu moramo podatke od nekod vzeti in jih po obdelavi nekam odložiti, da jih lahko kasneje spet uporabimo za naslednji proces. Umazano posodo vzamemo iz vhodnega pomnilnika, jo pomijemo, odložimo na odcejalnik (vmesni pomnilnik), pobrišemo in spravimo v omarico (izhodni pomnilnik). Pomnilnik je torej nekakšno skladišče in shranjevanje je tudi nekakšen proces.

Podatki potujejo po poračunalniku sem in tja po tako imenovanih **kanalih**. Umazana posoda gre z mize v pomivalno korito, od tam na odcejalnik in nato v omaro. Na prvi pogled je videti, da je promet podatkov samo enosmeren. Vendar ni tako. Po določenih kanalih je dovoljen promet v obe smeri. Če procesor za brisanje ugotovi, da skodelica ni dobro pomita, jo lahko vrne pomivalnemu procesorju po istem kanalu, po katerem je prišla. Če je kanal hkrati odprt v obe smeri, govorimo o **dupleks** prenosu, če je kanal izmenoma odprt zdaj v eno zdaj v drugo smer, govorimo o **polovičnem dupleks** prenosu, če pa je prenos podatkov mogoč le v eni smeri, kar pomeni da brisalec metlice za stepanje smetane ne more poslati nazaj, pa govorimo o **simpleks** prenosu. Več o osnovah elektronske obdelave podatkov in vseh prijetnih kuhinjskih opravilih prihodnjič.

Srečko Weber

Sinclair v znamenju simulacij

Igre meseca

Pobrskam po svoji zalogi kaset, iščem **Blast, Frankie goes to Hollywood** ali kaj podobnega. Kmalu ugotovim, da ničesar od tega nimam, najdem pa nekaj drugega: **Moon Creste**. Kdo od nas ni igral **Moon Creste** na igralnih avtomatih, preden nas je zasužnjil novi ljubimec Sinclair? Toda kaj, ko so pa vse igre na Spectrumu slabše kot na avtomatih! To je res, vendar z eno izjemo: **Moon Cresta** ni razen zvočnih učinkov nič slabša od izvirnika na avtomatih. Nedvomno sodi k najbolj znanim in najbolj tipičnim arkadnim igram. Vse, kar delaš, je to, da streljaš in se umikaš napadalcu. Tvoja raketa je sestavljena iz treh delov. Prvi del izstreli en naboj, druga dva pa dva. Oglejmo si najprej prvo stopnjo. Prikažejo se zvezde in tvoja ladja. Prst napeto počiva na gumbu tastature ali igralne palice. Prileti napadalci. Streljaš. Kadar kakšnega zadeneš, se pri priči razleti na dva dela. Napadajo te štirje napadalci, vendar jih je težko zadeti, ker se neprestano premikajo. Druga stopnja je na las podobna prvi, le da se napadalci vedejo bolj »živčno« in hitreje frfotajo sem ter tja. V tretji in četrti stopnji te napadejo rdeči in modri napadalci, ki povrhu vsega še spuščajo bombe. Sledi stopnja priključitve. En del tvoje ladje moraš priključiti na drugi del. Če uspeš, lahko streljaš s tremi oz. štirimi naboji naenkrat, ker je ladja sestavljena iz treh delov. Če te pri tem uničijo, zgubiš samo en del ladje. Po priključitvi spet pridejo napadalci-kometi, nato izvedeš še eno priključitev, seveda le v primeru, da imaš vse tri dele ladje. Če uspeš, lahko streljaš s petimi naboji. Na vrsto prideta še zadnji dve stopnji, ki sta skoraj enaki, ker se razlikujeta samo v agresivnosti napadalcev. Njihove ladje se spuščajo proti tebi in te skušajo zdrobiti. Ko uspešno opraviš tudi to nalogo, lahko začneš znova, pri tem pa se napadalnost povečuje. Grafika je lepa in hitra, problemov z atributi skoraj ni opaziti (zlasti na črno belem sprejemniku), zvok ni nič posebnega, igra je vredna, da jo imamo. Moje hackersko oko je vztrajno iskalo razliko med izvirnikom in Spectrumovo verzijo in jo tudi našlo. Na igralnem avtomatu se v prvih dveh stopnjah »oko« samo razpolovi na dva dela, če ga ne zadeneš dovolj hitro, tu pa ga razpoloviš le z zadetkom. Drugače pa bi zares mislil, da se igram z igralnim avtomatom — seveda, če se ne bi na zaslonu prikazal napis: »Choose joystick or keyboard« in če ne bi bil v moji sobi čist zrak brez cigaretnega dima. Igra je BIT HIT!

Ocene:

grafika	89
zvok	80
ideja	55
izdelava	95
možnost igranja	91
tržni uspeh	89
splošna ocena	91

Če ste se naveličali vseh **Flight Simulatorjev**, vam priporočam enega, ki ga najbrž še ne poznate: **Ski Star 2000**. Od drugih se razlikuje v tem, da simulira smučanje in ne letenje. Zamisel je dobra, izdelava tudi, le malo je prepočasen. Možnosti za spreminjanje pogojev na smučišču je kakih trideset. Na razpolago imaš vse od spreminjanja oblike očal do intenzivnosti sneženja, hitrosti ustavljanja, barv, smučarjeve velikosti, nagiba, za katerega se na različne načine obrnemo, in še in še. Izbiraš lahko med petnajstimi že narejenimi možnostmi.

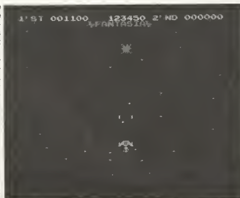
Ocene:

grafika	78
zvok	47
ideja	90
izdelanost	81
možnost igranja	86
tržni uspeh	87
splošna ocena	80

Po prvih uspehih, ki so jih prinesli **Hulk, Spiderman** idr., je programska hiša **Adventure International** prodrla naprej in izdelala novo avanturo **Gremlins**. Igra je strašna že na prvi pogled. Povsod te obkrožajo Gremlini, niti enega koraka ne narediš, če ne napišeš »Run down«. Rekli boste: »To ni nič posebnega«. Bolj nenavadno pa je to, da na nekaterih lokacijah Gremlini mahajo z rokami, predmeti izginjajo, ko jih pobereš ipd.

Ocene:

grafika	92
sistem	81
ideja	82
izdelanost	91
možnost igranja	93
tržni uspeh	89
splošna ocena	88



Lepa in hitra grafika: Moon Cresta

Ali ste za igro **Kokotoni Wilf** ali **Fall Guy**? Niste. No, ja, igri sta že malo stari, zato pa poznamo neko novejšo od Programske hiše **Elite**. To je **Air Wulf**. Igra je povsem drugačna. Ali pa ne. Vse **Elitine** igre so si bolj ali manj podobne. Njihova značilnost je, da vsebina ni grozovito zapletena, grafika je sijajna, poleg tega se često ponavlja. **Air Wulf** je igrica, v kateri nastopaš v obliki helikopterja. Prebiti se moraš skozi dvanajst zaslonov, polnih nevarnosti, od električnih pasov do radarjev. V treh zaslonih naletiš na kontrolno skrinjico, ki jo moraš zadeti, če želiš nadaljevati igro. Ko jo zadeneš, se ti odpre pot naprej. Navodilo za igranje: v drugem zaslonu pojdi do vrha in streljaj v zid. Ko ga prebiješ, se spusti za stopničko nižje in ponovi dejanje. To narediš trikrat, nato zletiš skozi zid. V tretjem ekranu trikrat prebij zid, nato pojdi dol v četrti zaslon. Tam zavij na desno, uniči skrinjico in se vrni na levo, kjer se pojavi odprtina. Ko prideš do konca leve strani, drži gumb za navzdol in na levo istočasno. Ko prideš v šesti zaslon, moraš takoj nadaljevati poševno navzdol, da te ne zadene laserski žarek. Posebna nevarnost je žarek, ki prihaja od spodaj. Počakaj, da izgine, nato pa hitro dol! Pojdi na desno, uniči skrinjico in se vrni. Žarek ni več sklenjen. Prebij zid in kreni navzdol. Uniči tretjo skrinjico in se vrni gor, nato kreni na desno, kjer je bila druga skrinjica. Sedaj je tam velika odprtina. Lahko nadaljuješ igro. Grafika je odlična, premikanje ne teče ravno po točkah. Od zvoka slišiš edinole motor svojega helikopterja in kaplje potu, ki ti curljajo na srajco.

Ocene:

grafika	94
zvok	86
ideja	80
izdelanost	81
možnost igranja	87
tržni uspeh	85
splošna ocena	80

Ocean nam je postregel z dvema novima programoma. To sta **Hunchback II** in **Kong Strickes Back**. Najprej poglejmo prvega. Programska hiša **Ocean** je po dveh letih obljubljanja izdala nadaljevanje **Hunchbacka** — tj. **Hunchback II**. Popravila je tudi prejšnjo napako. V prvi verziji je bilo vse preveč enako. Druga stopnja je bila malone posnetek prve. V tej verziji je sicer samo sedem različnih sob, ki pa se med seboj zelo razlikujejo tako po slikah kot po težavnosti. Ne manjka standardna **Oceanova** glasba na začetku igre.

Program je odlično izdelan in ni jasno, zakaj smo morali čakati nanj dve leti. Z igro **Kong Strikes Back** pa je drugače. Hočem reči, da me je malce razočaral. Že na začetku manjka prelepa glasba Oceanovih mojstrov. V igri, v kateri nastopa slavni Kong, sem si predstavljal vse nekaj drugega kot to, kar sem v njej videl. Nič ne rečem! Grafika je posrečena, toda samo na štirih zaslonih.

Hunchback II

Ocene:

grafika	90
zvok	93
ideja	65
izdelanost	87
možnost igranja	92
tržni uspeh	88
splošna ocena	86

Kong Strikes Back

Ocene:

grafika	89
zvok	51
ideja	69
izdelanost	77
možnost igranja	79
tržni uspeh	87
splošna ocena	71

Dobili smo še eno simulacijsko igro, tokrat z »resničnimi« letali. To je **Delta Wing** programske hiše **Creative Sparks**. O simulacijah na Spectrumu je bilo že toliko napisanega, da se bom omejil samo na to, kar druge simulacijske igre nimajo. To je v prvi vrsti izredna gibčnost letala, ki ga lahko obrnemo tako, da se ti skoraj zavrti v glavi. Odlično je prikazano sovražnikovo letalo. Zelo je nevarno in te lovi, kar je povsem drugače kot v igri **Fighter Pilot**, v kateri sovražnik samo beži. Krogla te lahko zadene celo v sprednje steklo, tako da vidiš luknje v njem. Komande so razmeroma preproste. V glavnem so take, kot v drugih simulacijah, od tistih, ki določajo položaj sovražnikovega letala, pa sta samo dve. Ena določi višino. Na instrumentu razbereš, ali si pod sovražnikom ali nad njim ali pa v isti višini. Drugi instrument pa kaže, v kateri smeri je sovražnik. Spet brez številk, uporabljaš samo grafikon. Ti si pika v sredini, sovražnik pa je pika zraven tebe. Oba radarja lahko delujeta na daljavo (en kvadratek) ali pa na bližino (desetina kvadratka). Cilj igre je uničenje sovražnikovega letališča. Toda pazite, da on ne uniči vašega! Letala lahko neomejeno razbijate, dokler imate še svojo bazo, od koder prihajajo nova. Ko je nimate več, se igra konča. Izbiraš med štirimi težavnostnimi stopnjami in različnim številom baz. Toda pazi! Za vsako sovražnikovo bazo imaš na razpolago samo eno bombo. Lepo je prikazano tudi premikanje tvoje roke v kontrolni kabini. Premika se sem ter tja, odvisno od tega, v katero smer naravnaš letalo. Prav tako se ti premikajo kolena. Igra mi je resnično všeč, mislim pa, da je narejena bolj po arkadnih kot po simulacijskih pravilih. Zame je to najboljša simulacijska igra, kar sem jih videl. Zasluži naziv BIT HIT.

Ocene:

grafika	93
zvok	81
ideja	43
izdelava	94
možnost igranja	97
tržni uspeh	96
splošna ocena	95

Programska hiša **Five Ways Software** nas je spet (ne)prijetno presenetila z igro **Run for Gold**. Simulacija prikazuje tekače na stadionu, toda izvedba igre je naravnost porazna. Ali tečeš naravnost ali pa po ovinkih, ki niso prav lepo izvedeni. Gumba za levo in desno ne smeš uporabiti, ker te takoj zanese ven s poti. Le figurica je dobro narejena. Zmagati ni težko, a težko je biti zadnji. Vsekakor bi igrica lahko bila boljša. V tem primeru bi rajši izbral **Decathlona**, kot pa to igro, v kateri igraš eno samo disciplino, a še ta je enolična in slabo izdelana.

Ocena:

grafika	61
zvok	42
ideja	51
izdelanost	48
možnost igranja	35
tržni uspeh	39
splošna ocena	38

Zdaj pa poglejmo še v naš »kotiček za hakerje«, namenjen vsem oboževalcem računalniških iger, ki ne znajo priti do njihovega konca in »krvavo« potrebujejo več življenj. Dobili smo prvo pismo z nasveti za igro **Flash**. **Rojc Matjaž** je opisal, kako zelo se je mučil z njo, pa ni dosegel nič. Na koncu je »vzel v roke« vse svoje hakerske sposobnosti in se lotil dela. Kmalu je odkril poke, ki je prav toliko zanimiv kot uporaben. Poke se glasi: POKE 59601,0. Če po naključju kdo ne bo znal pognati basic programa, naj si zapomni še tole: basic lahko spustite in napišete samo CLEAR 35000, nato naložite program z LOAD"" CODE in vpišete poke, program pa poženete z RANDOMIZE USR 59452. Drugače pa zadostuje, da »brek-ate« basic in vstavite poke med LOAD"" CODE in RANDOMIZE USR. Ko tako ali drugače poženete program, boste dobili nenavadno število življenj, ki se bo spremnilo, kadar boste »ubiti«, toda ne za eno navzdol, ampak malo navzgor in potem spet navzdol.

Sedaj pa še nekaj zanimivosti. Vsi, ki jih zanima igrica **River Rescue**, naj vpišejo POKE 33420,0 za igralce 1 in POKE 33452,0 za igralca 2. Tisti, ki pa bi rajši igrali kaj bolj arkadnega, vzemite **Defenderja** in naredite takole: spustite prvi basic del, nato napišite LOAD""CODE in poženite rak. Ko je igrica naložena, napišite še POKE 37815,255 ter RANDOMIZE USR 38658. Imeli boste 255 življenj in »pametnih« bomb.

Pojdimo na bolj igrane igre. Tisti, ki niste jedli sedem dni zaradi nepretrganega igranja **Jet Set Wilija**, sedaj gotovo ne jeste štirinajst dni zaradi **Sir Lancelota**. Dajem vam poseben recept. Napišite tale program:

```
1 REN življenje za Sir Lancelota
2 FOR a = 50000 TO 50024
3 READ n: POKE a, n: NEXT a
4 DATA 49,125,91,221,33
128,91,17,120,36,62,103
55,205,86,5,243,62,x,50 (x je število življenj od 1 do 127),
38,92,195,8,92
5 RANDOMIZE USR 50000
```

Nato naložite glavni del, ki je brez heakerja.

Za vse, ki vas je prejšnji mesec tako navdušil opis igre **Fantastic Voyage**, da ste ga brž kupili, a se zdaj zafrkavate, ker ne veste, ne kod ne kam, in vam je število življenj močno premajhno, pa še tole: Spustite prvi basic del in sliko. Napišite: CLEAR 30719: LOAD""CODE: POKE 53402,x (število življenj), ko pa se program naloži, natipkajte RANDOMIZE USR 53248. Imeli boste x življenj. Vendar pazite! Bolj ko se veča število življenj, manj pregleden postaja zaslon. Za vse, ki celo po tem navodilu ne bodo imeli uspeha (med temi sem bil tudi sam!), bom v naslednji številki pojasnil poke, s katerimi zlahka pridete do konca igre (če vas zares zanima, kaj je na koncu, mene nič več).

Spomnimo se starih dobrih časov, ko smo vsi sedeli pred televizijskim ekranom in »reševali« dekleta (ali pa fante) v ogromnem labirintu, polnem velikih mravelj. Spomnimo se še nadaljevanja te slavne igre, ki je sicer nekoliko manj slavno, toda v primerjavi z mnogimi drugimi igrami še kar uspešno. Gre za igro **Zombie-Zombie**, nadaljevanje **Ant Attacka**. Vsi ga poznate, pa vendar, kdo izmed vas ve, da sta v igri dva različna labirinta? Vse, kar je potrebno narediti, da pridete v drugi labirint, je tole: Naložite igro normalno, nato pojdite v glavni menu. Pritisnite P in nato še **CAPS Shift** (ne da bi P spustili). Pojavila se bosta dva narekovaja, v katera moramo vpisati SPACEMAN in pritisnite ENTER. Nato vpišete še xchg in pripravljeni smo na borbe v drugem labirintu Zombija. Če se želite vrniti v prvega, ponovite isti postopek.

Za konec pa še nasvet kako dobiti mnogo točk v igri **Bruce Lee**. M. R. iz Ljubljane nam je napisal, naj gremo po začetku igre v zaslon na desni in se postavimo na levi zgornji rob. Gumb za streljanje zalepimo s selotejpom in odidemo na malico. Kaj se zgodi? Če si na pravem mestu, ubiješ sproti vsakega napadalca, ki pride na ekran. Medtem ko malicaš, tepe tvoja figurica samuraje in pridobiva točke.

Preidimo k novim igram. Programska hiša **Ocean** marljivo izdeluje igre **Frankie goes to Hollywood, Street Hawk** in nadaljevanje **Decathlona,** ki se imenuje **Dalay Thompson's Super Test**, končala pa je igro **Ronald's Rat Race. Starion,** igra znane programske hiše **Melbourne House**, ki smo jo omenili že prejšnji mesec, je tudi končana, **A&F** pa obljublja igro **Chuckie Egg** 2. Tudi **Beyond** ne počiva. Muči se z igro **Shadow Fire.** Kaj bo iz vsega tega, pa bomo videli.

Lestvica desetih najboljših Spectrumovih iger za ta mesec (pravzaprav prejšnji, ker sem maja oddal rokopis) je taka:

Rang	Naslov igre	Programska hiša
1	Teshnitian Ted	Hewson Colsuntants
2	Everyone's A Wally	Mikro-Gen
3	Moon Cresta	Incentive
4	Bruce Lee	U.S.Gold
5	Sky Star 2000	Richard Sheperd
6	Fantastic Voyage	Quicksilva
7	Starion	Melbourne House
8	Gremlids	Adventure International
9	Air Wulf	Elite
10	Gyron	Firebird

Jernej Pečjak

Kako predstaviti simbole?

Premikajoče figurice ali sprites

Vsak malce izkušenejši uporabnik se pri delu z računalnikom slej ko prej sreča s problemom predstavitve različnih simbolov oziroma figur.

Večina računalnikov, kot tudi C-64 omogoča določevanje novega nabora znakov, saj na primer ob vklopu računalnika ne moremo izpisati recimo črke š. Poleg samega dodajanja črk oziroma njihovega spreminjanja (glej Bit 8), lahko uporabimo slednjo metodo tudi za oblikovanje novih grafičnih znakov, saj se mnogokrat pojavi potreba po znaku, ki ga na naši tipkovnici nikakor ne najdemo. Vendar pa ima opisani način oblikovanja lastnih znakov določene pomanjkljivosti, če bi ga želeli uporabiti tudi za prikaz vesoljčka ali pa hitrega avtomobila.

Kot vemo, je matrika znaka velika 8 krat 8 točk — 64 skupaj torej. To pa večinoma ne zadostuje za dovolj kvaliteten prikaz naše »umetnine«. Druga težava pa se javi, ko hočemo naš dirkalni avtomobil premikati po cesti — zanimivo, namesto da bi vozil enakomerno, ga kar premetava, kot da vozi po samih luknjah. Problem je jasen — znak lahko premaknemo najmanj za 8 znakov naenkrat — govorimo seveda o basicu!

Kaj sedaj?
No, k sreči naša štirinšestdesetletnica le ni samo »poslovni« računalnik, ampak omogoča tudi zelo zanimive grafične trike, ki se jih z malo truda in pazljivosti kaj hitro naučimo.

Govorimo seveda o figuricah, ali o SPRITES kot jih imenuje firma Commodore.

Kaj so figurice?

Na ekranu jih lahko opazujemo kot 24 krat 21 točk velike pravokotnike, v računalnikovem pomnilniku pa so predstavljene kot zaporedje 63 naslovov (+1 za zaokroženje).

Figurice si najverjetneje predstavljamo kot satovje. Ker pa se vse skupaj dogaja v računalniku, našega satovja ne bomo mogli polniti z medom, ampak s številkami. Na nek način bo to še lažje, saj bomo uporabljali le 0 in 1. Seveda, mnogi ste že uganili, zopet smo se znašli v dvojiškem sistemu.

Težaško delo?

Če nimamo programa za oblikovanje figuric, bomo morali uporabiti kos karo papirja. Narišemo pravokotnik širok 24 kvadratkov in visok 21. Sedaj pa vzemimo v roke svinčnik in pobarvajmo nekaj kvadratkov, da dobimo sliko. Ko smo sliko končali, jo je treba seveda še spremeniti v številke. Predstavljajmo si, da pomeni 0 prazen kvadratek in 1 pobarvanega. Z drugimi besedami nam torej kvadratek predstavlja en bit in ker vemo, da ima zlog (byte) 8 bitov, bomo sedaj spremenili po 8 sosednjih bitov v vodoravni vrsti v ustrezno desetiško število.

Postopek ni preveč zapleten, lahko pa postane malce zamuden, če nam bo dvojiška matematika delala preveč težav. Zato bomo prav na hitro omenili še to.

• Zlog — byte

	7	6	5	4	3	2	1	0
8 kvadratkov naše slike:								
dvojiško število:	0	0	0	1	1	0	0	0
desetiško število:								24

izračun: 0*2↑7 + 0*2↑6 + 0*2↑5 + 1*2↑4 + + 1*2↑3 + 0*2↑2 + 0*2 + 0*1 = 24

Na zgoraj opisani način spremenimo celoten pravokotnik v 63 števil v desetiškem sistemu — našo figurico smo torej kodirali.

• Zapis figurice v pomnilnik

Skupek 64 (63 + 1) števil, ki predstavljajo figurico, imenujemo tudi blok.

Če uporabimo za vpis blok 13 bomo torej vpisovali od naslova 832 (13*64) naprej.

Zapis je v basicu verjetno najenostavnješi s pomočjo ukaza READ, podatki pa se nahajajo v DATA stavkih.

• Naslovi podatkovnih blokov figuric

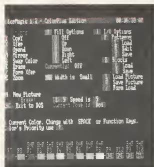
Naša slika se nahaja torej v 13 bloku. Sedaj pa se moramo odločiti, katera figurica bo uporabljala te podatke. Na voljo imamo 8 figuric, ki jih lahko sočasno postavimo na ekran. V naslove od 2040 do 2047 vpišemo bloke v katerih so podatki posameznih sličic.

2040 — figurica 0
2041 — figurica 1
2042 — figurica 2
2043 — figurica 3
2044 — figurica 4
2045 — figurica 5
2046 — figurica 6
2047 — figurica 7

Kot vidite, imamo lahko v našem pomnilniku shranjenih več sličic, kot jih lahko predstavimo naenkrat. Z ukazi POKE pa enostavno zamenjamo bloke iz katerih bomo brali podatke.

Tudi osemkratna predstavitev iste slike je zelo enostavna. Le v lokacije 2040 do 2047





je potrebno vpisati isto številko bloka, tako da bodo jemale vse figurice iste podatke in jih ni potrebno zapisati 8 krat.

• Položaj figurice na x in y osi

V registre video kontrolerja 0 do 16 oziroma naslove 53248 do 53264 vpisujemo koordinate naših figuric.

Ker ima naš ekran po y osi 200 točk, s tem ne bomo imeli težav. Zapletlo pa se bo pri x osi, saj je dolga 320 točk, ki jih ne moremo predstaviti z enim samim zlogom (največ 255). V tem primeru si moramo pomagati z registrom 16 oziroma naslovom 53264, kamor vpisujemo na ustrezno mesto najvišje bite (devete!) oddaljenosti po x osi.

reg	naslov	vsebina
0	53248	x os figurice 0
1	53249	y os figurice 0
2	53250	x os figurice 1
3	53251	y os figurice 1
4	53252	x os figurice 2
5	53253	y os figurice 2
6	53254	x os figurice 3
7	53255	y os figurice 3
8	53256	x os figurice 4
9	53257	y os figurice 4
10	53258	x os figurice 5
11	53259	y os figurice 5
12	53260	x os figurice 6
13	53261	y os figurice 6
14	53262	x os figurice 7
15	53263	y os figurice 7
16	53264	najvišji bit x osi vseh figuric

S spreminjanjem vrednosti teh registrov premikamo figurice po ekranu. Z x registri premikamo figurico levo in desno, z y registri pa gor in dol.

Primer vpisa najvišjih x bitov v register 16:

• Barva figuric

Tudi brez barve ne bo šlo. Za ta namen pa bomo uporabljali naslednje naslove:

reg	naslov	vsebina
39	53287	barva figurice 0
40	53288	barva figurice 1
41	53289	barva figurice 2
42	53290	barva figurice 3
43	53291	barva figurice 4
44	53292	barva figurice 5
45	53293	barva figurice 6
46	53294	barva figurice 7

Za barvo lahko uporabljamo vrednosti od 0 do 15.

• Povečava figurice

Kot veste ni nujno, da bo naša figurica na ekranu stalno iste velikosti, kot smo si jo sprva zamislili. Lahko jo povečamo po x ali po y osi, ali pa kar po obeh oseh naenkrat.

reg	naslov	vsebina
29	53277	povečava po x osi
23	53271	povečava po y osi

Uteži bitov ustrezajo številkam figuric.

• Prednost

Kadar želimo, da bi se ob stiku ozadja in figurice slednja skrila za znake na ekranu, moramo uporabiti naslov 53285. Za vsako figurico je zopet rezerviran en bit, katerega številka ustreza številki figurice.
Na tem mestu velja omeniti tudi prioriteto med figuricami samimi. Le-ta je pogojena s številko figurice in sicer imajo prednost figurice z manjšo številko.

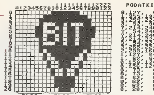
• Detekcija dotikov

Dodatne možnosti, ki jih glede animacije omogoča C-64, so za uporabnike, ki programirajo v basicu žal v mnogih primerih zaradi prepočasnosti prevajalnika (interpreterja) manj uporabljive, toda vseeno jih ne kaže zanemariti.
Registra 30 in 31 VIC čipa nam posredujeta informacije o trkih med posameznimi figuricami oziroma o trkih med figuricami in znaki na ekranu.

reg	naslov	vsebina
30	53278	trki med figuricami
31	53279	trki med figuricami in ozadjem

Zopet odgovarjajo številke bitov številkam figuric.
Zagnani programerji gotovo že premišljujejo, kako bo izgledala njihova nova akcijska igra s figuricami. Vse skupaj pa vas pred tem še vabim, da si ogledate tudi demonstracijski program, ki vam bo razjasnil še zadnje oblačke.
V tem programu je obdelanih nekaj tem, ki smo jih omenili v sestavku. Prikazano je branje podatkov za figurice in možnost uporabe istih podatkov za dve figurici. Poleg tega lahko vidite tudi, kako se figurico poveča, pobarva in premika na različne načine. Kot zanimivost je dodan test dotika dveh figuric, ki sproži spreminjanje barv okvirja.
Upamo, da ste prebrali dovolj zanimivosti, ki vas bodo verjetno zaposlile za precej časa in verjemite, po nekaj neprespanih nočeh, ali pa še prej — tudi uspeh ne bo izostal.

Peter Privšek




```
0 REM ******************************
1 REM *                            *
2 REM *      FIGURICE DEMO 1       *
3 REM *                            *
4 REM ******************************
10 PRINT"♥" : REM ZBRISI EKRAN
20 REM ******************************
21 REM *                            *
22 REM *          KAZALCI           *
23 REM *                            *
24 REM ******************************
26 V=13*4096 : L=2040 : D=832
30 REM ******************************
31 REM *                            *
32 REM * PREBERI PODATKE V BLOK 13  *
33 REM *                            *
34 REM ******************************
36 FOR I=0 TO 62
37 :  READ J : POKE I+D,J
39 NEXT
40 REM ******************************
41 REM *                            *
42 REM * BLOK PODATKOV ZA FIG 0 IN 1*
43 REM *                            *
44 REM ******************************
46 POKE L,13 : POKE L+1,13
50 REM ******************************
51 REM *                            *
52 REM * POSTAVI FIGURICI 0 IN 1    *
53 REM *                            *
54 REM ******************************
56 POKE V+21,3
60 REM ******************************
61 REM *                            *
62 REM * POVECAJ FIO 1 PO X OSI     *
63 REM *                            *
64 REM ******************************
66 POKE V+29,2 : POKEV+23,2
70 REM ******************************
71 REM *                            *
72 REM * POBARVAJ OBE FIGURICI      *
73 REM *                            *
74 REM ******************************
76 POKE V+39.7 : POKE V+40,14
80 REM ******************************
81 REM *                            *
82 REM * POBARVAJ EKRAN             *
83 REM *                            *
84 REM ******************************
86 POKE V+32,0 : POKE V+33,0
90 REM ******************************
91 REM *                            *
92 REM * POSTAVI Y KOORD FIG 1      *
93 REM *                            *
94 REM ******************************
96 POKE V+3,100
100 REM ******************************
101 REM *                            *
102 REM * PRETECI CEL X              *
103 REM *                            *
104 REM ******************************
106 FOR X= 1 TO 255 STEP 3
110 REM ******************************
111 REM *                            *
112 REM * PREMAKNI FIGURICI          *
113 REM *                            *
114 REM ******************************
116 POKE V,X
118 POKE V+1,100+(SIN((X-120)*3/120))*30
120 POKE V+2,X
122 POKEV+39,XAND15
124 REM ******************************
125 REM *                            *
126 REM *    TEST STIKA FIGURIC      *
127 REM *                            *
128 REM ******************************
129 IFPEEK(53248+30)=0THEN140
130 POKE53280,(B1+1)AND15:B1=B1+1
140 NEXT X : GOTO 100
200 REM ******************************
205 REM *                            *
210 REM *    PODATKI ZA SLICICO      *
215 PEM *                            *
220 REM ******************************
230 DATA 252,2,0,130,2,0,194,7,0,194,2
234 DATA 0,194,2,0,196,66,0,254,2,0
240 DATA 193,66,0,193,66,0,193,66,0,193
244 DATA 66,0,254,33,0,0,0,0,0,0,0,0,0,0
250 DATA 0,0,0, 0,0,0, 0,0,0, 0,0,0
255 DATA 0,0,0,0,0,0

READY.
```

Mehurčki, vendar ne milni...

Ponovno oživljena oblika magnetnega spomina

Kadarkoli omenimo računaniški spomin imamo običajno v mislih dve danes najbolj razširjeni obliki. To sta notranji spomin v obliki polprevodniškega integriranega vezja (RAM) in pa zunanji spomin v obliki magnetnega traku (na kolutu ali v kaseti) ali magnetnega diska ali pa diskete. Vsaka od teh oblik ima seveda svoje prednosti in pomanjkljivosti, za vse pa je značilno, da je hitrost s katero lahko računalnik prečita shranjene podatke v obratnem sorazmerju s ceno za shranitev teh podatkov.

Vse bolj pa se je začela uveljavljati ponovno oživljena oblika magnetnega spomina — magnetni mehurčki (bubble memory). V poznih šestdesetih letih so mnogi predvidevali da bodo ti mehurčki nadomestili takratno »core memory« — miniaturne kolobarčke iz magnetne snovi, ki so shranjevali bite.

Tako kot na disketi so tudi v »mehurčkastem spominu« nosilci informacije mikroskopska magnetizirana območja, ki predstavljajo posamezne bite z vrednostjo 1, medtem ko ostalo območje predstavlja bite z vrednostjo 0. Tu pa se podobnost z disketo neha. Mehurčki niso shranjeni na plastiki ampak na čipu, tako kot na primer RAM. To pomeni da ni gibajočih se delov. Za razliko od RAM spomina pa mehurčki ne izginejo, ko se prekine dovod električnega toka, kar je v veliko primerih izrednega pomena. Podatki so na takem čipu veliko hitreje dosegljivi kot na disketi, čeprav znatno počasneje kot na RAM. Čip tudi ni narejen iz silikona, kot so RAM čipi, ampak iz kristalov granata.

Mehurčki pravzaprav niso povsem pravi izraz — na čipu ni nikakršnih izboklin, ki bi bile podobne mehurčkom. Namagnetena območja so valjaste oblike in so pravzaprav skupine atomov, katerih magnetni poli so urejeni. Magnetna polja takih območij so običajno neurejena, ko pa je granat primerno razrezan v tanke ploščice pa se polja uredijo tako, da so pozitivni in negativni poli na zgornji ali pa na spodnji površini ploščice. Os magnetnega polja je torej pravokotna na površino. (risba 1)

Ti magnetni mehurčki sami po sebi niso stabilni, rastejo ali pa se manjšajo dokler ne izginejo. Da bi bili uporabni, jih je treba stabilizirati s pomočjo zunanjega magnetnega polja. Pod vplivom tega polja se mehurčki uravnovesijo pri velikosti približno dve tisočinki milimetra. Pod mikroskopom taka območja izgledajo kot mehurčki — od tod tudi ime.

Območja s pozitivnim polom na zgornji površini so seveda posamezni biti informacije, ki imajo vrednost 1. Kakor dolgo je zunanje magnetno polje stabilno, so tudi mehurčki stabilni ter ne rastejo niti se ne zmanjšujejo.



Če je informacija shranjena na magnetnem traku, mora le ta potovati preko čitalne glave, ki najde in prečita željene podatke. Do informacije na disku ali disketi mora računalnik priti s premikom čitalne glave in z vrtenjem diska. Pri mehurčkih pa je zadeva povsem drugačna. Magnetna območja — mehurčki — potujejo po čipu in tako pridejo do čitalne glave, ki prečita podatke. Za to je treba čip spraviti v območje vrtečega se magnetnega polja, kar se doseže z dvema tuljavama in primernim električnim tokom skozi njiju. Mehurčki potujejo v zaporedju po kovinskih vodih, ki so vdelani v čip.

Podatke na RAM čipu lahko računalnik prečita neposredno, ker ima vsak bit svoj naslov, medtem ko je do mehurčkov na tak način nemogoče priti. Do željenega bita informacije je mogoče priti le tako, da potujejo vsi mehurčki pred njim mimo čitalne glave. Če je na čipu shranjenih milijon bitov in je tisti ki nas zanima nekje na polovici verige, se mora vsa vrsta premakniti za polovico dolžine. Povprečna hitrost premika mehurčkov na granatnem čipu pa je približno 200.000 korakov v sekundi. V tem primeru bi preteklo okoli dve sekundi in pol, preden bi prava informacija prišla do čitalne glave. Čeprav se na prvi pogled zdi to dokaj hitro, pa je veliko prepočasi za kakršenkoli računalnik.

Rešitev problema, kako do volj hitro priti do željene informacije leži v oblikovanju kovinskih vodil, po katerih mehurčki potujejo do čitalne glave. Le ta so položena v obliki zelo izdolženih elips položenih ena poleg druge. Okoli teh elips je zunanja zanka, ki se jih dotika na obeh straneh (risba 2) in ki vodi do čitalne glave. Originalni biti so seveda še vedno shranjeni na notranji zanki. Na podoben način se novi podatki vnesejo z zunanje zanke v katerokoli izmed notranjih. Ta izmenjava mehurčkov na vhodni ali izhodni strani med zunanjo in vsemi notranjimi zankami lahko poteka sočasno.

Tako se prvotnih 500.000 korakov (povprečje za čitanje z milijon bitnega čipa) zmanjša na vsega 1.500 korakov in čas z dveh in pol sekund na približno 40 milisekund. Čeprav je ta čas okoli 1000 krat večji kot pri RAM, je dva do štirikrat manjši kot pri disku.

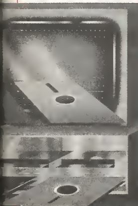
Resnična prednost mehurčkov je v količini podatkov, ki se lahko shrani in pa kako dobro ostanejo ti podatki shranjeni. Napake pri tej vrsti spomina so zelo redke. Intel navaja za svoj 128 kilo bytni čip da je možnost da bi uporabnik dobil bit z nepravilno vrednostjo ena v 10.000.000.000.000.000. Pri



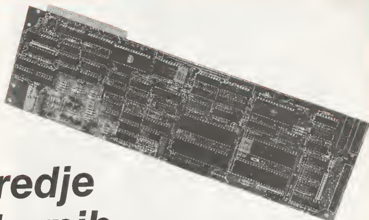
vsakdanji uporabi to pomeni neka-ko eno napako vsakih sto let! Ta zanesljivost je še nadalje poudar-jena s tem, da ostanejo podatki ne-dotaknjeni tudi v primeru če pride do prekinitve električnega toka.
Zunanje magnetno polje, ki ohra-nja mehurčke pri življenju je vzpo-stavljeno s pomočjo trajnih ma-gnetov, ki so neodvisni od prego-rele varovalke ali slučajno iztak-njenega električnega kabla, ki na-paja računalnik. Če se dovod elek-tričnega toka do čipa prekine, se ustavi vrteče magnetno polje, ki premika mehurčke. Tok podatkov se ustavi, to pa je tudi vse.
Ta lastnost ohranjanja podatkov neodvisno od električnega toka je seveda zelo pomembna pri vsaki uporabi, kjer morajo podatki ostati shranjeni v vsakem primeru. Tak spomin je seveda tudi prenosen, saj se lahko čip vzame iz računal-nika, pošlje drugam in ko se po-novno vstavi v računalnik, bodo podatki še vedno na njem in to tak-šni, kot so bili. V primerjavi z RAM čipi imajo mehurčki še drugo pred-nost, ki je pomembna pri vsako-dnevnem zmanjševanju velikosti vseh računalniških delov. Običa-jen granatni čip shranjuje na pribli-žni površini šestih kvadratnih cen-timetrov 128 Kb podatkov, med-tem ko enakovredni RAM shrani le 32 Kb.
Cena je zaenkrat še slaba stran granatnega spomina. Ko je Intel najavil prve 128 Kb čipe leta 1979, je bila cena 2500 dolarjev za čip. Do danes je cena sicer padla na 300 dolarjev za tak čip, kar pa je v primerjavi z disketami še vedno zelo veliko. Vendar pa je tehnolo-gija na tem področju še razmero-ma nova, po drugi strani pa proiz-vodnja še ni prav masovna. Cena mehurčkov je odvisna od tega, ko-liko je prodanih. S povečano proiz-vodnjo se bo cena zmanjšala. Vendar pa so ovire za razvoj prav gotovo velike. Medtem ko dela na razvoju silikonskih čipov v ZDA okoli 50.000 inženirjev, se jih z me-hurčki ukvarja le kakih 200.

Pa še ena stvar je, ki nekoliko zavi-ra prodor mehurčkov. Ni namreč še povsem jasno kaj naj bi nado-mestili, notranji spomin (RAM) ali zunanji spomin (diskete). Na tre-nutni stopnji razvoja še vedno zao-stajajo ali po zmogljivosti ali po do-stopnosti za obema vrstama spo-mina na njunem področju. Vseka-kor pa so že našli pot v različne računalnike. Trajnost in trpežnost jih postavlja kot naravno izbiro za računalnike v tovarniški proiz-vodnji, kjer bi vsak diskovni pogon zelo hitro obnemogel zaradi prahu ali vibracij. Manjše mere so jih postavile na drugi strani v elitni razred prenosnih računalnikov, na primer v mikroračunalnik Com-pass firme Grid Systems.
Z najavljenim 512 Kb čipom v letu 1986 se bo uporaba mehurčkaste-ga spomina prav gotovo še razširi-la. Zahteva po hitrem in zaneslji-vem spominu se bo vedno večala in magnetni mehurčki imajo z mil-nimi skupno samo ime.

Sašo Novak







Zaporedje enostavnih opravil

Oblikovalci mikroprocesorjev so pri določanju nabora ukazov zelo pazljivi, saj mora biti izvajanje obsežnih operacij zaporedje enostavnih opravil, od katerih vsak predstavlja en ukaz iz dobro oblikovanega seta ukazov. Spodaj prikazana tabela je kratek povzetek nabora ukazov pri mikroprocesorju 6510. Ukazi so razdeljeni po podobnih si skupinah. V drugi tabeli je prikazan nabor ukazov, urejen po številčnem vrstnem redu strojnih kod v heksadecimalni obliki.

Da bi bila tabela nabora ukazov razumljiva, sledi razlaga okrajšav, uporabljenih v tabeli:

Registri:

A
Akumulator

X
Indeksni register X

Y
Indeksni register Y

PC
Programski števec

SP
Kazalec na sklad

P
Register stanj z biti označenimi kot sledi:

7 6 5 4 3 2 1 0 ◄— ŠTEVILKE BITOV

| | V | | B | D | I | Z | C | ◄— REGISTER STANJ (P)

REZERVIRANO ZA RAZŠIRITVE (sedaj ni v rabi)

Stanja:

S
Predznak ali negativen rezultat

V
Prekoračitev dolžine bajta (overflow)

B
Prekinitev

D
Decimalna aritmetika

I
Onemogočanje prekinitev

Z
Ničelen rezultat

C
Prenos (carry)

Simboli v kolonah pri stanjih:

(prazno)
izvajanje ne vpliva na stanje

x
izvajanje vpliva na stanje

o
izvajanje briše stanje

1
izvajanje postavi stanje

6
izvajanje reflektira šesti bit spominske lokacije

7
izvajanje reflektira sedmi bit spominske lokacije

Ostali simboli:

nasl
osem bitov absolutnega ali osnovnega naslova

(nasl + 1, nasl)
naslov sestavljen iz vsebin spominskih lokacij »nasl« in »nasl + 1«

nasl 16
šestnajst bitov absolutnega ali osnovnega naslova

podatki
8 bitov direktnih podatkov

pomik
8 biten označen premik naslova

labela
16 biten absoluten naslov skoka (jump) ali skoka na podprogram (jump-to-subroutine)

PC(HI)
zgornjih osem bitov programskega števca

PC(LO)
spodnjih osem bitov programskega števca

PP
drugi bajt dvo- ali tro-bajtne strojne kode

qq
tretji bajt tro-bajtne strojne kode

()
vsebina spominske lokacije označena znotraj oklepajev. Na primer, (FFFE) predstavlja vsebino spominske lokacije FFFE šestnajstiško: (nasl 16 × X) predstavlja vsebino lokacije, naslovljene s prištevanjem vsebine registra X k nasl 16; (SP) predstavlja vrednost na vrhu sklada (vsebina spominske lokacije, naslovljene s skladovnim kazalcem).

(())
indirektno naslavljaje: vsebina bajta v spominu, naslovljenega z vsebino spominske lokacije, ki je označena znotraj dvojnega oklepaja. Na primer, ((nasl × X)) predstavlja vsebino spominske lokacije, naslovljene preko preindeksiranega indirektnega naslavljanja.

+
seštevanje — bodisi brezznakovno binarno ali BCD seštevanje, odvisno od stanja za decimalno aritmetiko.

—
binarno ali BCD odštevanje, ki se izvaja s prištevanjem dvojiškega komplementa odštevanca k minuendu.

—
eniški komplement količine, označene pod črto. Denimo, A predstavlja komplement vsebine akumulatorja; C predstavlja komplement vrednosti stanj prenosa.

&
logični IN

∨
logični ALI
⊻
logični ekskluzivni ALI
←
podatki, prenešeni v smeri puščice.
Mnemoniki ukazov
Stolpec UKAZ je namenjen mnemonikom ukazov (LDA, JSR, TAX) in operandom, če so.

Stalni del ukaza v zbirniku je izpisan z velikimi črkami, spremenljivi del (neposredni podatki, naslovi, priključne točke-labele) pa z malimi črkami.

Če ima mnemonik več tipov operandov, je vsak operand izpisan v svoji vrstici pod mnemonikom kot na spodnjem primeru:

STY
nasl
nasl, X
nasl 16
pomeni:
STY $75
STY $60,X
STY $4726

Izvorne kode ukazov
Za bajte ukazov brez variant, so izvorne kode predstavljene kot dve šestnajstiški številki (npr.: C5). Za bajte ukazov z variantami, so izvorne kode predstavljene kot osem binarnih številk (npr.: 01011×01).

V tabeli je prikazana izvorna koda in dolžina ukaza v bajtih za vsako varianto ukaza posebej.
Čas izvajanja ukaza

Čas je izražen v številu period sistemske ure. Stvaren čas izvajanja pa se lahko izračuna z deljenjem danega števila period s hitrostjo ure. Na primer, ukaz ki zahteva 4 periode ure, ki niha s frekvenco 2 MHz se bo izvajal 2 mikrosekundi.
Stanja

Zastavice stanj so shranjene v registru stanj (P), ki je bil predstavljen v prvem delu (skica 3 na strani 23).

NAČINI NASLAVLJANJA PRI MIKROPROCESORJU 6510

Mikroprocesor 6510 nudi enajst osnovnih načinov naslavljanja:

1 — spomin podatki neposredno
2 — spomin absolutno ali neposredno, ne stran nič
3 — spomin stran nič (neposredno)
4 — naznačeno ali inherentno
5 — akumulator
6 — pre-indeksno posredno
7 — pro-indeksno posredno
8 — stran nič indeksno (tudi osnovna stran indeksno)
9 — absolutno indeksno
10 — relativno
11 — posredno

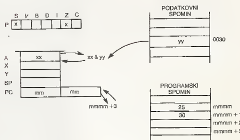
Katera metoda je dovoljena s katerim ukazom je prikazano v tabeli nabora ukazov.

SPOMINI Z NEPOSREDNIMI PODATKI
Pri tej obliki naslavljanja je eden od operandov v bajtu, ki sledi bajtu strojne kode. Neposreden operand je označen z znakom ≠.

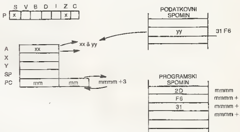
Na primer:
AND ≠$08
zahteva od zbirnika da generira ukaz IN med vrednostjo 08 šestnajstiško in vsebino akumulatorja.

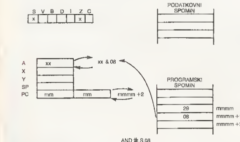
Biti, ki izberejo IN ukaz	Biti, ki izberejo Neposredno naslavljanje z enim operandom v A

SPOMIN NEPOSREDNO
Ta oblika naslavljanja uporablja drugi — ali drugi in tretji (če ni na strani nič) — bajt ukaza za določanje naslova operanda v spominu. Oblika s stranjo nič se uporablja kadar se izraz uporablja kot operand v ukazu zreduciranem na vrednost med 00 in FF šestnajstiško. Na primer:
AND $30
zahteva od zbirnika da generira ukaz IN med vrednostjo v spominski lokaciji 0030 šestnajstiško in vsebino akumulatorja.



Absoluten način (ne preko strani nič) je podoben, le da naslov operanda zaseda dva bajta. Na primer:
AND $31F6
zahteva od zbirnika da generira ukaz IN med vrednostjo v spominski lokaciji 31F6 šestnajstiško in vsebino akumulatorja.





	7	6	5	4	3	2	1	0
1 bajt	0	0	1	0	1	0	0	1
2. bajt	0	0	0	0	1	0	0	0

Šestnajst biten naslov je shranjen prvo z zadnjimi osmimi biti (na spodnjem naslovu), katerim sledi prvih osem bitov (na zgornjem naslovu). Enaka tehnika je uporabljena pri 8080, 8085 in Z80 mikroprocesorjih, pri 6800 pa je ravno obratno.

NAZNAČENO ALI INHERENTNO
Pri tej obliki ni potrebno naslavljanje za izvajanje ukaza. Tipičen primer inherentnega naslavljanja je CLC (briši prenos) in TAX (prenesi register A v register X).

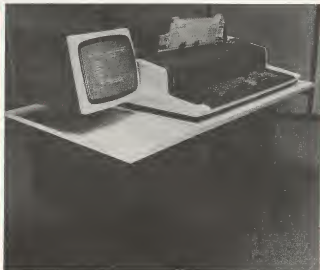
NASLAVLJANJE AKUMULATORJA
Pri tej obliki ukaza operira s podatki v akumulatorju. Na mikroprocesorju 6510 so taki ukazi ASL (aritmetični pomik v levo), LSR (logični pomik v desno), ROL (vrti levo skozi prenos) in ROR (vrti desno skozi prenos).

Janez Majdič

(Se nadaljuje)

Birostroj Maribor

V korak z računalništvom



Eden izmed Birostrojevih najpomembnejših izdelkov je mali poslovni sistem RGB 110, ki je namenjen za distribuirano avtomatsko obdelavo podatkov v različnih gospodarskih organizacijskih enotah. Zasnovan je kot samostojna enota za obdelavo podatkov. Njegova konfiguracija je prilagojena potrebam tržišča za celovito obdelavo posameznih programskih paketov
Nekaj tehničnih podatkov:
- *centralno enoto sestavlja 8-bitni mikroračunalnik (Z 80), ki ga vodi operacijski sistem BROS; spomin znaša od 16 do 64 Kbyte*
- *tipkovnica je sestavljena iz alfanumerične, numerične in funkcijske ter dela popolnoma na elektronskem principu*
- *zaslon ima ekran s kapaciteto 1920 znakov (24 vrst × 80 znakov), katerega diagonala meri 31 centimetrov*
- *tipkalnik robotron 1152 dela na principu marjetice (Daisy-wheel) s hitrostjo tiskanja 40 znakov v sekundi; vseh znakov je 96, največja širina izpisa pa je 210 znakov v vrsti*
- *formularska tehnika: leporello — transporter papirja, obdelava posameznih formularjev, naprava za konto kartice tip 1161 in naprava za avtomatsko vodenje posameznih obrazcev tip ASF 176/576*
- *zunanjo pomnilno enoto sestavljata dve enoti floppy diska za 8-colske diskete kapacitete 256 KByte (IBM format)*

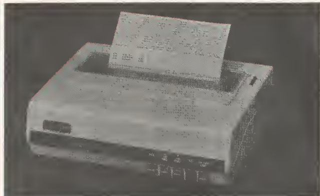
Lahko bi rekli, da je osnovna značilnost mariborskega Birostroja dinamičen razvoj, ki prinaša s seboj zelo pomembne spremembe. Ta delovna organizacija deluje namreč na razgibanem, v prihodnost stremečem in tudi negotovem ter do konca neopredeljenem področju računalništva, zato vedno znova išče poti za uspešno vključevanje v mednarodno delitev dela, ne da bi pri tem zapostavila domače gospodarske cilje. Lani se je na osnovi kooperacijske pogodbe z Gorenjem bolj intenzivno usmerila v izdelavo računalniških poslovnih sistemov in se preimenovala v Birostroj, delovno organizacijo za proizvodnjo malih poslovnih sistemov Maribor. Vsaka sprememba pa zahteva čas, še zlasti, če je vezana na tako bogato, kar tridesetletno tradicijo, kot jo ima Birostroj. Rezultatov zato ne morejo pričakovati kar čez noč. Takole pravi generalni direktor Mirko Hartner: »Sprememba je zamišljena tako, da še naprej ohranjamo in razvijamo vse, kar je pozitivnega, odstranimo pa prepreke, ki zavirajo programirani razvoj naše delovne organizacije.«

Birostroj je torej z novim letom startal tudi v novi obliki s štirimi osnovnimi dejavnostmi: razvojem, proizvodnjo, trženjem in storitvami. Najpomembnejša sta seveda razvojni oddelek, ki ima izjemen položaj med vsemi drugimi in v politiki razvoja delovne organizacije, ter proizvodnja, ki še zaostaja za zahtevami kupcev, čeprav je pri nas že mogoče opaziti njihovo slabšo kupno moč. Proizvodnja temelji na združevanju prostih tehnoloških zmogljivosti Birostrojevih poslovnih sodelavcev, zato doslej ni zahtevala dodatnih vlaganj v tehnologiji. Glede na vse višjo zahtevnost končnih izdelkov pa se bo morala delovna organizacija zavzeti za prototipno proizvodnjo in končno kontrolo izdelkov, pri čemer bo treba natančno določiti naložbe v svojem ter v okviru tehnoloških procesov sovlagateljev, denimo Gorenja, Iskre itd. Prav na področju vlaganja v proizvodnjo in razvoj pa je bistvenega pomena konkurenčnost izdelkov na tujih tržiščih. Dosedanja ocena proizvodnje dokazuje, da jih zelo dobro sprejema Jugoslavija in vzhodne države, medtem ko zaradi cene in kvalitete ne morejo uspeti na zahodnem trgu. Gre zlasti za mala poslovna sistema RGB 110, ki ga izdelujejo v Mariboru, RGB 130, ki prihaja iz ptujske Elektronike ter matrične tiskalnike RGB 105. Rešitev za navedene težave bo najbrž treba iskati v večjem združevanju aplikativne elektronike in tesnejšem povezovanju z domačimi delovnimi organizacijami, denimo z Iskro-Mikroelektroniko.

Znano je, da ima pri sodelovanju s tujimi podjetji Birostroj že precej izkušenj. Za najbolj uspešno se je izkazalo poslovno tehnično sodelovanje s kombinatom Robotron. Iz programa le-tega je mogoče pri Birostroju kupiti računalniške sisteme in periferne sklope, video terminale, najsodobnejše tiskalnike in drugo računalniško opremo. Delovno organizacijo zastopa tudi avstrijski Gould, ki ponuja pri nas merilne instrumente, osciloskope, analizatorje logike ter digitalne termopisalce z vsemi servisnimi storitvami. In če smo že pri servisiranju, velja omeniti, da

Birostroj popravlja računalnike iz programa Commodore, ki jih prodaja ljubljanski Konim, in sicer v svojem sektorju precizne mehanike na Gregorčičevi ulici v Mariboru. S tem pa vseh dejavnosti te mariborske »računalniške« delovne organizacije še ni konec. Izdeluje namreč tudi servisno opremo, instrumente in električne sklope v majhnih serijah, strojne agregate in druge naprave na podlagi kooperacijskih odnosov, popravlja pisarniško opremo in fotokopirne stroje, kupuje in prodaja nova in rabljena mehanografska sredstva ter rezervne dele zanje, programira in uvaja računalniško podprte informacijske sisteme v gospodarske in druge organizacije združenega dela na osnovi sodobne informacijske strojne in programske opreme, izobražuje delavce drugih delovnih organizacij na področju operaterstva in programiranja in še bi lahko naštevali. Tudi v naslednjih letih ima Birostroj kopico načrtov, od nadaljnjega tesnega povezovanja z domačimi institucijami in spremljanja razvoja v tujini, kar bo omogočilo proizvodnjo še bolj kakovostne pisarniške in sistemske računalniške opreme, vse tja do organiziranega izobraževanja svojih delavcev. Vse to bo mariborski Birostroj še utrdilo med najpomembnejšimi jugoslovanskimi delovnimi organizacijami, ki izdelujejo in prodajajo računalniško opremo. Zanj in za vse poslovne sodelavce z jugoslovanskega področja je bilo zato izredno zanimivo posvetovanje na Bledu, ki je potekalo od 29. do 31. maja in na katerem so predstavili svoje najnovejše izdelke. To je bila priložnost za nove dogovore in sodelovanje in s tem uvajanje najsodobnejše računalniške tehnologije tudi pri nas.

(rš)-Bep



Birostroj je svojo ponudbo malih poslovnih sistemov pred nedavnim obogatil z novim, pomembnim dosežkom — matričnim tiskalnikom RGB 105, ki je izhodna enota za izpis podatkov na različne obrazce, neskončne formularje ali papirnate role. Na razpolago sta dva modela: 105/80 in 105/132. Zaradi preprostega menjavanja interfacea in seta znakov je tiskalnik prilagojen za različna področja uporabe. Sklada se z večino računalniških sistemov, ki jih uporabljamo pri nas.

Tehnični podatki:

- *hitrost izpisa je 100 znakov v sekundi*
- *širina izpisa je 80, 100 ali 120 znakov v vrsti pri modelu 105/80 in 132, 165 ali 190 pri modelu 105/132, odvisno od gostote zapisa*
- *mogoče so različne vrste pisave: običajna, široka, podčrtana, kurzivna in kombinacije*
- *neposredno nastavljanje iglic omogoča semigrafiko*
- *število kopij je pri obeh modelih 1 + 2*
- *največja širina obrazca je pri modelu 105/80 252 milimetrov, pri modelu 105/132 pa 406 milimetrov*
- *pisalni trak je enak kot pri pisalnem stroju*
- *mere: model 105/80 — 370 × 280 × 130 milimetrov*
  model 105/132 — 580 × 285 × 135 milimetrov
- *teža: model 105/80 — 6 kilogramov*
  model 105/132 — 8,5 kilograma

Mali poslovni sistem RGB 130 je namenjen distribuirani avtomatski obdelavi podatkov v gospodarstvu in negospodarstvu. Je samostojna obdelovalna enota z možnostjo nadaljnjih obdelav podatkov na višjih sistemih v off-line ali in-line povezavi.

Konfiguracija:

— centralna enota

Za opravljanje funkcij malega poslovnega sistema RGB 130 se uporabljajo moduli 8-bitnega mikroračunalnika K 5120, ki jih vodi operacijski sistem SIOS. Spominski moduli so zgrajeni v RAM in ROM tehniki in jih je možno dograditi za standardno konfiguracijo do kapacitete 64 K byte, po potrebi pa dopolniti do maksimalne kapacitete 112 K byte. Obstaja možnost vgraditve CMOS RAM, ki v primeru izpada električne energije omogoča shranjevanje informacij.

— nosilec podatkov

Za zapis podatkov so vgrajeni:
— gonila za 8" gibljive diske (floppy) z enostranskim zapisom enojne gostote in kapacitete 256 K byte, formatirano v IBM formatu (single side — single density) ali
— gonila za 8" gibljive diske (floppy) z enostranskim zapisom dvojne gostote in kapacitete 512 K byte (single side — double density).

— prenos podatkov

Za daljinski prenos podatkov se uporabljata dva različna interfacea, V.24 in 20 mA zanka (IFSS). Prenos podatkov je lahko sinhron ali s hitrostjo 50 do 9600 baudov, kar omogoča povezavo malega poslovnega sistema RGB 130 z višjimi računalniškimi sistemi. Kot komunikacijski protokoli se uporabljata protokola AP 62/64, BSC 1 in BSC 3.

— razširitev sistema

Mali poslovni sistem RGB 130 lahko dopolnimo z malim poslovnim sistemom RGB 120, ki rabi izključno kot zajemalna enota v off-line povezavi. Prednost takšne povezave je povečanje števila obdelav uporabniških programov na malem poslovnem sistemu RGB 130. Sistem je mogoče dopolniti z dodatnim tiskalnikom, kar omogoča večjo hitrost obdelave podatkov.

— aplikacije

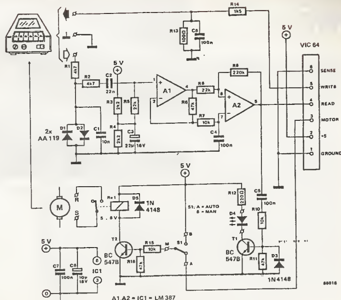
Mali poslovni sistem RGB 130 je zgrajen tako, da odpira nove možnosti programiranja in uporabe v praksi, predvsem pa prilagajanje potrebam uporabnika, ne samo danes, ampak tudi v prihodnje, ko se bo želel priključiti na višje računalniške sisteme.



Kasetni vmesnik za C-64 in VC-20

Poskusite sami

Zgodi se, da se kakšno vezje zgradi iz čiste potrebe. Tako je tudi z vmesnikom, ki ga bomo opisali.
Kot je razvidno iz vezja na sliki 1, ne potrebujemo preveč elektronskih elementov, da bi si napravili kar dober vmesnik: dva operacijska ojačevalnika in nekaj drobiža. Vendar najprej nekaj obrobnih informacij. Lastniki commodorjev gotovo vedo, da prihajajo podatki iz računalnika v obliki pravokotnih signalov z amplitudo okoli 5 Vss. Enake oblike impulzov pa morajo priti tudi na vhod računalnika. Poleg tega pa ima kasetni priključek še poseben vhod (6) s katerim računalnik preverja ali je na kasetniku pritisnjena tipka »Play«. Če je to res, se aktivira izhod (3) »Motor«. Tega uporabimo za krmiljenje kasetnikovega motorja. Na to možnost se še povrnemo.



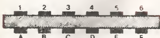
Vmesnik s podrobnostmi

Prilagoditev izhodnih signalov računalnika na normalen kasetofon je zelo enostavna. Preko napetostnega delilnika R13/R14 zmanjšamo izhodnih 5Vss na 200 mV, kondenzator C6 pa odstrani »umazanijo«. S tem je izhodni signal že »čist« za zapis na kasetniku. Torej lahko programe že »varno« shranimo. Pri »nalaganju« programov je potrebno malo več opravil.
Signal, katerega odda audio kasetnik preko DIN priključka, ali preko priključkov zvočnika je po eni strani »ne več tako« pravokotne oblike, po drugi strani pa preko DIN priključka s premajhno amplitudo, okoli (200... 300 mV). Torej moramo signale ojačati in regenerirati, če pridejo iz DIN izhoda, ali pa omejiti in regenerirati, če pridejo iz zvočnika. Omejitev je enostavna. Opravita jo diodi D1 in D2. Zatem se signal z operacijskim ojačevalnikom A1 okoli 6 krat ojača. Z uporoma R3 in R4 dosežemo polovično pogonsko prednapetost (2,5 V) na vhodu tega ojačevalnika A1. Kasetni izhodni signal je simetrično naložen na to prednapetost. Operacijski ojačevalnik A2 deluje kot Schmitt-Trigger oscilator, tako da daje na izhodu čiste pravokotne signale 5 Vss.
Sedaj lahko prečitamo program preko »Read« vhoda v računalnik. Ko nastopijo podatki, postane T1 vedno prevoden, če je na izhodu operacijskega ojačevalnika A2 logična 1. Vsled tega zagori led dioda D4. Ker se pri čitanju na izhodu ojačevalnika A2 menjajo logične 0 in 1 s precejšnjo frekvenco imamo občutek, da led dioda neprestano gori. Kar s pridom uporabimo za iskanje začetkov programov na kasetniku.
Sedaj pa h krmiljenju motorja. Če prestavimo stikalo S v položaj »A« (za avtomatiko), potem krmili tranzistor T2 rele R1 preko »Motor« izhoda iz računalnika. Pri tem moramo računalniku simulirati, da je pritisnjena tipka »Play« v kasetofonu. Za ta namen smo prej omenjeni vhod »Sense« (6) vezali na maso. Pri commodorjevem kasetniku pa »Play« tipka opravi isto. V našem primeru je najenostavneje, če ta vhod vežemo na maso. Pogonsko napetost za vmesnik posreduje računalnik preko priključkov (1, 2), tako da ne potrebujemo usmernika za napajanje vmesnika.
Povezave.
Predno lahko uporabimo vezje, moramo vzpostaviti potrebne povezave. Slika 2 prikazuje kasetni priključek računalnikov C64, VC-20. Odgovarjajoče vtikače kupimo v trgovini. Na omenjeni plošči s tiskanim vezjem in v stikalnem načrtu so označeni vsi priključki. S tem ne bi smelo biti večjih problemov za vzpostavitev pravilnih povezav. Kvečjemu še R in S priključka za relejno stikalo. Predvideno je, da se priključita na vhodno stikalo kasetnika. Če to ni možno, prekinemo en dovod do motorja kasetnika in vmes povežemo relejno stikalo.

Sestava, vgradnja in uporaba

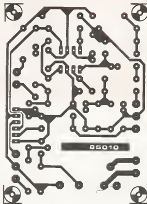
Slika 3 prikazuje ploščico s tiskanim vezjem. Odločitev o tem ali bo vmesnik v posebnem ohišju ali pa vgrajen v sam računalnik prepuščamo vsem, ki se bodo gradnje lotili sami. Pripominjamo samo to, da povezave ne bi pre-

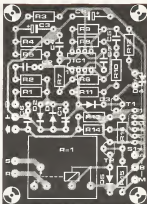
- **KONEKTOR RAČUNALNIKA**



Skica 2: 1. masa (GND), 2. napajanje (5V), 3. motor, 4. read (čitanje, 5. write (zapis), 6. sense (indikacija pritisnjenosti tipke PLAY).

segle 1 metra, kar obenem preprečuje kakršnekoli probleme z linijami.
Tudi o uporabi ne bi preveč govorili. Vse je opisano v commodorjevem priročniku. Kajti z našim vmesnikom je audio kasetnik enako uporabljan kot commodorjev podatkovni kasetnik, z razliko, ki jo omogoča stikalo Sl. Preklapljamo lahko med avtomatičnim krmiljenjem preko računalnika in ročnim posredovanjem. To zadnje izkoriščamo za hitro previjanje traku, in za hitro iskanje programov s pomočjo vgrajenega števca na kasetniku.





Skica 3.

Še zadnje opozorilo: če bi računalnik med nalaganjem programov opozoril na napako, je najverjetnejši vzrok nepravilna nastavitev glasnosti kasetnika. Predvsem se to dogaja pri nalaganju iz tujih kaset. Takrat pač poskusimo z novo nastavitvijo glasnosti. Vendar po naših izkušnjah s tem vmesnikom celo pri »Turbo« nalagalniku redko pride do takega primera.

Pokaži, kaj znaš — kviz za vsako glavo

Že nekaj mesecev so v Ameriki izredno priljubljene igre »Pokaži, kaj znaš!« pod tem in drugimi imeni, »Igra znanja«, »Spekter znanja« itd. V eni igri je 2000, 6000 tisoč ali celo 10000 vprašanj in odgovorov.
Zelo priljubljena je »Trivia Fever« za IBM PC/jr in commodore 64. Pripravljajo tudi inačico za Evropo. V kvizu lahko sodeluje do 8 ljudi, ki morajo izbirati med petimi področji: tehnika in znanost, zemljepis, zgodovina, šport, film in zabava, slavne osebe, narava in živali. Pred igro določimo, na koliko vprašanj je treba odgovoriti v izbranem času. Zmagovalec mora svojo zmago potrditi še z »odgovorom igre«.
»Trivia Fever« lahko igramo z računalnikom ali brez njega. 3000 vprašanj in odgovorov (ki so razdeljeni v tri težavnostne stopnje) je natisnjenih tudi v priloženi knjigi. Če se igramo z računalnikom, ta naključno izbira vprašanja, če se igramo brez njega, pa imamo vprašanja, napisana na vrtljivem papirnatem kolutu. Ali je takšna igra v krogu prijateljev bolj prijetna kot pogovor ali ne, to boste presodili sami. Res pa je, da bi njeni sestavljalci lahko bolje izrabili grafične možnosti računalnika. Tudi presojo o tem, ali so vprašanja najbolje izbrana, prepuščamo vam, dragi bralci in bralke. Program trenutno prodajajo le v Ameriki za 39.95 dolarja.



Video učitelj deli čebelice

Govoreči osebni računalnik je lahko bolj učinkovit kot učitelj: to potrjuje sistem za učenje branja in pisanja, ki so ga začeli uporabljati v ZDA. Imenuje se *Writing to Read* (Do branja s pisanjem), ki je dobil dovoljenje za dvoletno uporabo v 22000 vrtcih in osnovnih šolah. »Sistem je zelo učinkovit učni pripomoček in močno vzpodbuja učenje,« pravi Richard Murphy, psiholog z inštituta Education Testing Service, ki je ocenjeval ta sistem. Po mnenju tega inštituta je Writing to Read »primeren« za šolsko uporabo.
Računalnik kaže učencem različne stvari in zraven izgovarja imena. Gre za 30 besed, ki vsebujejo 42 fonemov (glasov) angleškega jezika. Učenci ae potem učijo pisati na tipkovnico, hkrati pa jih sintetizator zvoka uči pravilne izgovorjave. V programu so tudi igre in pravljice. IBM, ki je dal na razpolago računalnike trdi, da stroški opreme (računalniki, programi, knjige, minikasete, igre, magnetofoni, slušalke) niso višji od 35 dolarjev na učenca letno.

Najboljši imajo umetniško žilico

Kakšni so vrhunski programerji?

končnega izdelka, prinaša tudi povsem otipljivo korist. Nič čudnega, saj marsikateri programer zavestno uporablja tudi estetska merila, za preizkušanje programov.

»Če najdemo napako, tedaj tudi z obliko nekaj ni v redu,« takšen je na primer značilen odgovor programerja. »Ko imam takšen program pred sabo, takoj vem, da je OK. To vidim na prvi pogled. Morda lahko to primerjamo s sposobnostjo nekaterih ljudi, da v besedi-



Kadar govorimo o programerjih, si ponavadi predstavljamo ljudi, ki s hladno logiko, korak za korakom in strogo razumsko pišejo računalniške programe. V resnici pa ni tako. Vrhunski programerji delajo intuitivno. V tem so zelo podobni vrhunskim matematikom, ki jih pri delu prav tako vodijo nezavedni impulzi.

Strokovnjaki tudi ugotavljajo, da programiramo veliko lažje, hitreje in kvalitetneje, kadar smo v posebnem duševnem stanju, ki se razlikuje od normalnega budnega stanja in nekateri se že leta ukvarjajo s pojavom »nezavednega« programiranja. Marsikakšen velik projekt se je ponesrečil, ker je prišlo do velikih težav, ki pa niso bile tehnološke narave, temveč težave s človeškim faktorjem. Zato se je zelo koristno podrobneje seznaniti s tem, kako delajo uspešni programerji.

Vrhunski programerji so pri programiranju izredno močno koncentrirani. Navadnim programerjem je program seznam vseh ukazov, iz katerih sestoji. Z vrhunskimi programerji je drugače. Enemu je »program tridimenzionalna stopničasta struktura, okoli katere se lahko premikam in stopam po njej. Ko iščem napako v programu, se v duhu postavim poleg njega, tako da dobim pravo perspektivo«.

Še bolj zanimivo je, kako ta programer program preizkusi kar v glavi. »Takrat postanem pika in kot pika tečem skozi program — skozi vse zanke in preskoke. Vsak korak opravim v glavi. Potem sem prepričan, da je program stoodstotno pravilen. Če na ta način ne najdem napake, pomeni, da je vse lepo in prav.«

Kako pa dobi pravo perspektivo, ko pregleduje program? »Takrat pravzaprav obstajam dvojno. Enkrat sem pika zunaj programa, drugič gibljiva pika znotraj programa.«

Podobne izjave so dali tudi številni drugi programerji, pa tudi strokovnjaki iz ZDA, ki že leta preučujejo to nenavadno dejstvo. Začetnik raziskav »človeškega faktorja« je ameriški računalniški strokovnjak Gerald Weinberg, ki je že pred dvanajstimi leti izdal knjigo »Psihologija računalniškega programiranja« (the Psychology of Computer Programming).

Kaj so pokazale dosedanje raziskave na tem področju? Programiranje je, tako kot matematika, le deloma strogo razumska dejavnost. »Kdor dela le z razumom«, pravi eden od strokovnjakov, »je daleč od pravih virov ustvarjalnosti«. Vrhunski programerji so si tako ali drugače utrli pot do teh virov, bolj ali manj zavedno v svoje delo vključujejo nezavedno (nezavedno je v tem primeru ustreznejši izraz kot podzavestno). Vrhunski programerji imajo močno razvit estetski čut, ni jim dovolj, da program deluje, temveč mora biti tudi lep in eleganten na pogled. Dobra mehka oprema mora zadovoljevati tudi potrebo po umetniški uveljavitvi.

Zelo zanimivo je, da tovrstno estetsko prizadevanje poleg jasnosti, pravilnosti in optimalnosti lu takoj najdejo pravopisne napake, ne da bi ga sploh brali; besede, ki niso pravilno napisane, izstopajo, ker so videti čudne in smešne.« Nekateri »delajo« celo med spanjem. »Kadar nekaj časa zastonj iščem napako,« pravi eden od teh umetnikov, »tedaj vem, da je vse skupaj najbolje prespati. Zjutraj takoj najdem napako«.

Intuitivnega dela se lahko naučijo tudi navadni smrtniki. Osvoboditi se morajo stvari, ki jim ne dajo, da bi v polni meri razvili ustvarjalnost. Za konec si zastavimo še vprašanje, čemu naj bi ljudi sploh učili, da bi programirali intuitivno? Če ne zavoljo drugega, tedaj zato, da bi na področje, ki je danes še vse preveč podrejeno strojem, vnesli več človečnosti in zadovoljstva, kajti človek je vendarle veliko več kot le stroj za programiranje.

Sinclairjevci, pozor!

Če kakšnega programa morda kot po kakšnem čudežu še nimate, je zdaj prilika za poceni nakup v Nemčiji: Space Riders, Reversi, Psion Chess, Flight Simulation in Chequered Flag stanejo le 5.95 mark. Znižali so tudi cene programov za ZX 81. Tiskalnik »Alphacom 32« za Spectrum in ZX81 stane zdaj le 198 mark in je torej najcenejši Sinclairjev tiskalnik.

Dynamicsova vesela palica

Hajka za sovražniki ali volkodlaki in dirka za svetovno prvenstvo v Monzi sta uspešni le, če ima mojster v roki dobro igralno palico. Dynamics je nemški trg zasul z zares poceni veselo palico, ki ji je ime Ascom in stane 29,95 mark.

Na mizo se prisesa s štirimi plastični gumbi in ima štiri gumbe za streljanje. Dva sta na ročaju, dva na okrovu. Stikalna mehanika omogoča določanje osmih smerih.

Dynamics je prav tako izboljšal staro dobro Competition Pro in jo opremil z mikro stikali. Za TI99 in Schneider izdeluje adapter, ki omogoča priključitev dveh igralnih palic na en vhod, ki stane 25,50 DM. Ker je veselje tokrat zelo poceni, objavljamo naslov za informacije: Dynamics, Grosse Beckerstr. 11, 2000 Hamburg 1.

ACT sodeluje s Tandyjem

ACT in Tandy sta ustanovila podjetje za prodajo na evropskem tržišču. Ker ima Tandy v ZDA težave s prodajo mikroračunalnikov, gotovo upa, da bo imel več sreče s prodajo actijevih marelic.

Razširitve s Tajvana

Tajvanski proizvajalec računalnikov Multitech, ki izdeluje tudi z IBM PC združljiv osebni računalnik, je izdelal pet razširitvenih kartic za IBM PC. Gre za razširitev glavnega pomnilnika za 384 KB, adapter za barvno grafiko, ki ima dovolj prostora tudi v IBM XT, vmesnik za asinhrono komunikacijo, kartico z dvema serijskima vmesnikoma in enobarvni zaslonski adapter, ki ima tudi serijski vmesnik.

Gora disket je zmeraj višja

Proizvodnja dodatne opreme za računalnike doživlja nesluten razmah. Poznavalci računajo, da bo letos proizvodnja disket narasla za približno 60 %. Še leta 1981 je bila rast 33,3 %, lansko leto 47,1 %. Ker se trenutno za svoj delež v prodaji od za letos predvidenih 40 milijonov kosov poteguje skoraj 100 proizvajalcev, lahko pričakujemo precejšnje znižanje cen. Že ob koncu lanskega leta se je cena za dvostranske diskete z dvojno gostoto znižala na 40 DM za 10 disket.

Pomnilnik za Macintosh

Tecmar je izdelal kombiniran pomnilnik z vmesnikom za Applov macintosh. Trdi disk ima kapaciteto 10 MB, zamenljivi pa 5 MB.

Disketa za 3,3 MB

Kodakova disketna enota z zmogljivostjo 3,3 megabyta je zdaj naprodaj. V Nemčiji jo prodaja BN System Engineering v Münchnu. Disketna enota je zelo tenka (slimline) in ima vgrajen mikroprocesor. Superzmogljiva disketna enota je primerna tudi kot podsistem za priključitev na osebne računalniške sisteme.



900 PROGRAMOV ZA C-64 na kasetah in disketah prodam. Nastja Škopac, 61000 Ljubljana, V Murglah 103.

PROFESIONALNO D'KTRONICS TIPKOVNICO, SPEKTRUM 48K in programe ugodno prodam. Telefon (061)321-900.

PROGRAMI ZA SPEKTRUM PO UGODNI CENI! Za 10 naročenih programov popust. Možna tudi menjava. Pišite za brezplačni katalog. Rajko Orel, 61234 Mengeš, Prešernova 21.

NAJCENEJŠI PROGRAMI ZA ZX SPEKTRUM od 20 do 60 dinarjev. Posneti na kasetofonu Nordmende. Poleg tega več kot 60 POKOV za različne igre po nizkih cenah. Pišite za brezplačen katalog na naslov: Jože Žitnik, 61217 Vodice, Krvavška 17.

KUPIM PROGRAME ZA COMMODORE 64. Telefon (063)751-084, Goran.

KING SOFTWARE, igre, devpak, beta basic, CV, rtty, sstv... Mijo Kovačević, 63212 Vojnik, Cesta talcev 2.

PROGRAME ZA COMMODORE 64, 16, 116 in Plus 4 prodajam. Spisek brezplačen. Milutin Miljuš, Tuk 2, 43212 Rovišće.

UGODNO PRODAM SISTEM spektrum 48 K, kasetofon in TV-aparat Trim. Informacije na telefon (064)50-838, Matjaž.

V uvožene tiskalnike in računalnike »vgrajujem« črke č, š, ž in druge znake. Prevzemam tudi servisiranje tiskalnikov. AOP Igor Ferjan, telefon (061) 571-482 ves dan in (063) 28-609 popoldan. 03684

PRODAM COMMODORE 64 z vgrajeno reset tipko, datasete in programi. Andrej, telefon (061)346-194.

PROGRAMER'S REFERENCE GUIDE — original in prevod v srbohrvaščino prodam. A. Dvoršak, Posavskega 5, 61000 Ljubljana.

LIGHT-PEN ZA C-64

Novo!!! Najnovejše!!!
Light pen 400 ND
program na kaseti — 500 ND
program na disketi — 1000 ND

Naslov: LIGHT-PEN
pri borštu 5
61210 Lj.-Šentvid
Tel.: (061)51-148

M.C.P. software. Prodaja in zamenjava programov za spektrum. Nizke cene (30 — 40 dinarjev). Brezplačen katalog! Jelenc Dušan, 64260 Bled, Partizanska 31. Telefon (064)77-671, po 17. uri.

ZA CBM 64 — prodam najboljše programe po nizkih cenah, katalog brezplačen. Telefon (061)316-265. (032293)

PROGRAME ZA C-64 PRODAM IN MENJAM. Cena 50 dinarjev. Kupim disketno enoto za C-64. Ponudbe na naslov: Ivan Benkovič, Novi trg 24, 61240 Kamnik.

JB KLUB NUDI NAJNOVEJŠE IN NAJATRAKTIVNEJŠE PROGRAME za commodore 64. Posnamemo v 24 urah. Kakovost zajamčena. Žrebanje. Pošljemo katalog in vse informacije. Telefon (061)579-238 ai 579-296.

ZA COMMODORE 64 PRESNEMAVAM PROGRAME NA DISKETO ALI KASETO. Pišite za seznam. Andrej Sevčnikar, Otiški vrh 5, 62373 Šentjanž, Dravograd.

PROGRAME ZA SHARP MZ-700 KUPIM ali zamenjam za programe računalnika tip MSX. Andrej Česnik, 64000 Kranj, Šiškovo naselje 38.

ZX SPEKTRUM — NAJBOLJŠA PONUDBA PREK 200 PROGRAMOV kot so: Everyone's A Wally, Bruce Lee, Blue Max... Prodam ali zamenjam. Pišite za brezplačen katalog. Boris Mihajlović, Obrtniška 3, 61230 Domžale.

Nasveti — zamenjave

Oglas v tej rubriki je brezplačen

RABIM NASVETE ZA CBM 64. Zamenjujem programe za spektrum. Pišite: Rengeo Dejan, Šalovci 3, 69204 Šalovci.

MENJAVA PROGRAMOV ZA »GALAKSIJO«! Javite se na naslov Janko Novak, 68290 Sevnica, Šmarska cesta 38. Priložite spisek svojih programov. Programe menjam izključno na kasetah.

NAJVEČJO IZBIRO SOFTWARA (programi, knjige, navodila) za spektrum menjam. Svoje spiske pošljite na naslov: Milan Vrca, 11070 Novi Beograd, Zarija Vujoševića 79.

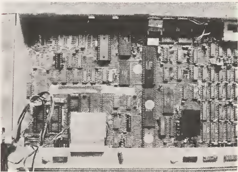
ZA COMMODORE 64 MENJAM igre in druge koristne programe. Tit Uhan, VP 8164/17-2, 62002 Maribor.

FANCY SOFT — prvi klub v Jugoslaviji za izključno zamenjavo programov za CBM 64. Klub želi kupiti program Prolog ali Mikroprolog za CBM 64. Pišite za spisek ali pa se včlanite. Braslav Erpačić, Vladimirja Nazore 8, 43404 Šp. Bukovica, Bušetina.

ZA TI-99/4A menjam programe in literaturo. Jozo Udovičič, 68270 Krško, Gubčeva 3.

NEMŠKA NAVODILA ZA ZX SPEKTRUM zamenjam za angleška. Božo Pirc, Vlahovićeva 33, 61000 Ljubljana, tel. 443-024.

MENJAM NAVODILA ZA IGRE in uporabne programe za ZX spektrum. Borut Španovič, 61000 Ljubljana, Cesta v mestni log 70/130B.



Izkušnja preteklosti, okus sedanjosti...



ronhill®

vrhunska moška kozmetika

Ronhill Red

Skrbno izbrane najkvalitetnejše francoske dišave združene v eleganten parfumski akord. Z vašo novo dišavo Ronhill Red boste pritegnili pozornost ženskega sveta. Enaka dišavna nota spremlja bogato izbiro kozmetičnih izdelkov za moške Ronhill Red.

Ronhill Black

Markantna, aromatična francoska dišava z nevsiljivo noto tobaka in ambre se bo najbolje prilegla odločnim, aktivnim moškim. Lahko ste prepričani, da bo tudi vaša izbranka zadovoljna z vašim okusom.



Ronhill Brown

Dišavni kompoziciji linije Brown daje najmočnejšo značilno prisotnost naravnega mošusa. Privlačen, moderen in atraktiven.

KRKA kozmetika

Vozilo: Tehniški muzej Slovenije

»DELOVNI DAN SE PRIČNE S PRITISKOM NA TIPKO IN S PRIJAZNIM POZDRAVOM NA ZASLONU »ZDRAVO, PARTNER«. NEPREGLEDNI KUPI PAPIRJA SO IZGINILI, ARHIV JE UREJEN IN SHRANJEN NA MAJHNIH, PRIROČNIH DISKETAH. SAMO TRENUTEK IN ŽE JE NA ZASLONU POSLOVNA ZGODOVINA, PISMA, TRENUTNI POSLOVNI REZULTATI ALI NAPOVED BODOČNOSTI, IN ŠE IN ŠE, SKRATKA VSE, ČEMUR DANES PRAVIMO AVTOMATIZACIJA PISARNIŠKEGA POSLOVANJA.«

Moj Partner

DOBRO JE, DA IMAM SVOJEGA PARTNERJA

Moj PARTNER skupaj s programi FILEPLAN, MICROPLAN, MEMOPLAN in TISKTIP so pripravljeni tako, da jih lahko uporabljajo vsi, četudi sami niso doma na področju računalništva.

FILEPLAN je nepogrešljiv pripomoček za enostaven vnos podatkov in oblikovanje preglednic, ki so nujno potrebne za hitre poslovne odločitve.

MICROPLAN je sistem planiranja na finančnem področju, ki ste ga že dolgo čakali. Omogoča analizo »KAJ SE ZGODI, ČE?«, načrtovanje in spremljanje poslovnih dogodkov, izpis poročil in drugo.

MEMOPLAN je prijateljsko preprost in učinkovit urejevalnik besedil. Njegove zmožnosti so tolikšne, da omogoča sočasno oblikovanje petih dokumentov. Program TISKTIP je posebna verzija za tiste, ki želijo sami oblikovati in pripravljati besedila neposredno za fotostavek v tiskarni.

Moj PARTNER ima 128 KB notranjega pomnilnika, disketno enoto (1 MB) in disk (10 MB) ter priključek za tiskalnik. Lahko imate svojega PARTNER-ja z dvema disketnima enotama, povežete ga lahko dodatno tudi z lepopisnim ali matričnim tiskalnikom.

Vsem uporabnikom računalnikov moj PARTNER je na voljo razvejana vzdrževalna služba v mestih širom po Jugoslaviji in šolanje v izobraževalnih centrih ISKRE DELTE. Novost so enodnevni brezplačni seminarji o uporabi PARTNER-ja v Ljubljani, Beogradu, Sarajevu in Skopju. Namenjeni so v prvi vrsti kupcem ter našim bodočim partnerjem.

Moj Partner


Iskra Delta

Sami se prepričajte v resničnost navedenih trditev! Izpolnite kupon, napišite svoj naslov ali pa preprosto priložite svojo poslovno vizitko in vse skupaj pošljite na naš naslov

p. p. 581
61001 LJUBLJANA

Naslov ______

želim ponudbo — želim dodatne informacije — želim vabilo za seminar

KUPON

6



Prilogo Zeleni BIT želimo »napolniti« s programi, ki jih bodo sestavljali bralci sami. Od števila teh programov je odvisen tudi obseg priloge. Prepričani smo, da je med lastniki računalnikov veliko tistih, ki so že ali pa še bodo izdelali različne programe, ki bi bili zanimivi tudi za druge bralce Bita.

Pričakujemo tudi vaše programe, ki bodo seveda ustrezno honorirani — od 2.500 do 13.000 dinarjev.

Enostavnejše izpisovanje programov

Čitljiv in razumljiv izpis

Kot ste že gotovo opazili, smo vam v tej številki vaše in naše revije pri nekaterih programih postregli z novim načinom izpisovanja programov.
V upanju, da bodo tako izpisani programi preglednejši in enostavnejši za »mukotrpno« vtipkavanje, smo se odločili za 45 znakov dolgo vrstico
Ker so izpisi namenjeni seveda vam, vas naprošamo, da nam sporočite vaše želje in pripombe, tako da bomo slednjič s skupnimi močmi pripravili zares čitljiv in razumljiv izpis.

Sedaj pa bo že kar čas, da tudi opišemo delovanje programa oziroma izgled izdelka programa z imenom 5T
Prva stvar, ki jo uporabnik opazi, je izpis vrstičnih številk programa. Izpis je formatiran, in sicer na 5 mest V kolikor vrstična številka ni petmestna, jih program izpiše s sprednjimi ničlami Poleg številk je v vrstici še največ 39 znakov.

Posebnost predstavlja tudi izpis kontrolnih znakov, ki se precej razlikujejo od običajnega izpisa, ki ste ga vajeni iz Bitovih prejšnjih številk
Namesto kontrolnega znaka, izpisanega v negativu se izpiše sedaj angleški izraz v oglatih oklepajih. Priznati moramo, da smo sprva poskusili poiskati za vse kontrolne znake slovenske sinonime, toda niso se nam ravno najbolje posrečili, tako da smo se morali vrniti k originalnemu komodorjevemu označevanju. Namesto npr. 12 zaporednih izpisov kontrolnega znaka za pomik v levo se sedaj izpiše [12LEFT]. Posebno poglavje so presledki Vsakdo se gotovo spominja, kako je pri zadnjem prepisovanju programa štel presledke in jih moral ponavadi popravljati še po izvajanju. Upamo, da sedaj s tem ne bo več težav saj se presledki izpišejo podobno kot kontrolni znaki, kateri je dodano še število ponavljanj.

Kontrolni znaki:

BARVE:

BLK	— črna barva	CTRL + 1
WHT	— bela barva	CTRL + 2
RED	— rdeča barva	CTRL + 3
CYN	— tirkizna barva	CTRL + 4
PUR	— vijolična barva	CTRL + 5
GRN	— zelena barva	CTRL + 6
BLU	— modra barva	CTRL + 7
YEL	— rumena barva	CTRL + 8
ORG	— oranžna barva	COMMODORE + 1
BRN	— rjava barva	COMMODORE + 2
LT/RED	— svetlo rdeča barva	COMMODORE + 3
GRY1	— barva siva 1	COMMODORE + 4
GRY2	— barva siva 2	COMMODORE + 5
LT/GRN	— svetlo zelena barva	COMMODORE + 6
LT/BLU	— svetlo modra barva	COMMODORE + 7
GRY3	— barva siva 3	COMMODORE + 8

FUNKCIJSKE TIPKE:

[F1]	funkcijska tipka številka ena
[F2]	funkcijska tipka številka dva
[F3]	funkcijska tipka številka tri
[F4]	funkcijska tipka številka štiri
[F5]	funkcijska tipka številka pet
[F6]	funkcijska tipka številka šest
[F7]	funkcijska tipka številka sedem
[F8]	funkcijska tipka številka osem

OSTALO:

LOCK	izklopi preklop med grafičnima načinoma
UNLOCK	vklopi preklop med grafičnima načinoma
RET	tipka RETURN
BUSINESS	poslovni način grafike
DOWN	premik slednika za mesto dol
RVS	prehod v negativni način predstavitve
HOME	vrnitev slednika v zgornji levi kot
INST	vrivanje znakov
RIGHT	premik slednika za mesto desno
SPACE	presledek
[S/RET]	RETURN s SHIFTOM
[GRAPHICS]	grafični način znakov
[UP]	premik slednika za mesto gor
[RVS/OFF]	konec negativnega načina predstavitve
[CLR]	brisanje ekrana
[INST]	vrivanje znakov
[LEFT]	premik slednika za mesto levo

V prihodnji številki bomo program tudi objavili in boste tudi svoje programe lahko izpisovali na novi način

TOP-CAT vam bo pomagal

COMMODORE 64

Programi, oziroma imena programov se v direktorij vpisujejo po vrstnem redu zapisovanja. Sčasoma se disketa napolni in direktorij kaj hitro postane nepregleden. Program TOP-CAT vam v tem trenutku priskoči na pomoč. Vse programe, ki se nahajajo na disketi vam razvrsti po abecednem redu.
Uporaba je zelo enostavna, saj je potrebno disketo, ki jo želimo urediti, le vstaviti v disketno enoto VC-1541, vpisati RUN in pritisniti tipko RETURN — za vse ostalo pa poskrbi program.
Najprej se nam na ekranu izpiše sedanji direktorij, potem pa sledi razvrščanje in direktorij se nam izpiše v abecednem vrstnem redu. Na koncu se novi direktorij še zapiše na disketo in postopek je gotov.
Program je priporočljiv predvsem v primeru, ko smo disketo napolnili in bi jo radi zaščitili ter jo v prihodnje uporabljali le za branje.
V kolikor boste za razvrstitvijo zopet vpisovali programe na disketo, se bodo ti vpisovali kot običajno po kronološkem vrstnem redu.
Program pravilno prepisan je več kot priporočljivo uporabljati za sortiranje na poskusno disketo, ki jo napolnimo z različnimi imeni.

```
00010 REM  ******************************
00011 REM  *           TOP-CAT           *
00012 REM  ******************************
00013 REM  *    RAZVRSCENI DIRECTORIJ    *
00014 PEM  ******************************
00015 REM  *      EDVARD . MAJ 1985      *
00016 PEM  ******************************
00017 PEM  +-----------------------------+
00019 REM  +                             +
00020 REM  +      OPIS SPREMENLJIVK      +
00021 REM  +                             +
00022 REM  +-----------------------------+
00023 REM  PN$ - POLJE PREBRANIH IMEN
00024 REM  SP  - POLJE PREBRANIH SLEDI
00025 REM  SE  - POLJE PREBRANIH SEKTORJEV
00027 PEM  DR% - STEVILKA DISKETNE ENOTE
00028 REM  I,J - SPLOSNI STEVEC
00029 REM  N$  - NICELNI ZNAK [CHR$(0)]
00030 REM  NN$ - NIZ NICELNIH ZNAKOV
00031 REM  POL - POLOZAJ BUFFER POINTERJA
00032 REM  E$ - PREBRANI ZNAK IZ VMESNIKA (BU
      FFERJA)
00040 REM  SP% - SLED
00041 REM  SE% - SEKTOR
00044 REM  PN% - STEVEC PRI VPISU IMEN
00100 PEM  +-----------------------------+
00101 REM  +                             +
00102 REM  +       DOLOCITEV POLJ        +
00103 REM  +                             +
00104 REM  +-----------------------------+
00105 DIM PN$(224):PEM  IMENA PROGRAMOV
00106 DIM SP(29):REM  SLEDI (TRACK)
00107 DIM SE(29):REM  SEKTORJI
00110 N$ = CHR$ (0)
00111 FOP I = 1TO 30
00112 :NN$ = NN$+N$
00113 NEXT
00120 REM  +-----------------------------+
00125 PEM  +                             +
00130 REM  +         POSTAVITEV          +
00135 REM  +                             +
00140 REM  +-----------------------------+
00150 FL% = 0:DR% = 0
00160 SO = 1:SP% = 18:SE% = 1:IF FL% THEN SP%
       = 39
00170 REM  +-----------------------------+
00175 PEM  +                             +
00180 REM  +    BRANJE DIPEKTOPIJA       +
00185 REM  +                             +
00190 REM  +-----------------------------+
00200 OPEN 15,8,15,"I"+STR$ (DP%):GOSUB 570
00210 OPEN 2,8,2,"#":GOSUB 570
00220 GOSUB 610:REM  IZPISI GLAVO
00230 PPINT# 15,"U1[SPACE]";2;DR%;SP%;SE%:SP(
      SN) = SP%:SE(SN) = SE%
00240 GOSUB 560:REM  PREBERI ZNAK
00250 SP% = ASC (E$):REM  VPISI SLED
00260 GOSUB 560:REM  PREBERI ZNAK
00270 SE% = ASC (E$):POL = 2:REM  VPISI SEKTO
      P.
00280 PRINT# 15,"B-P";2;POL:REM  POSTAVI KAZA
      LEC VMESNIKA (BUFFER POINTER)
00282 NN = NN+1:REM  POVECAJ STEVEC VPISOV
00290 FOR I = 1TO 30
00292 :GOSUB 560
00300 :PN$(NN) = PN$(NN)+E$
00302 NEXT
00310 :IF ASC (LEFT$ (PN$(NN),1)) = 0 THEN PN
      $(NN) = "":NN = NN-1:DE = DE+1:GOTO 350
00320 :FOR I = 19TO 5STEP -1
00322 :IF MID$ (PN$(NN),I,1)<>CHR$ (160)
       THEN I = 5:GOTO 340
00330 :PN$(NN) = LEFT$ (PN$(NN),I-1)+"[SPACE]
      "+PIGHT$ (PN$(NN),30-I)
00340 NEXT
00342 PRINT ,NNMID$ (PN$(NN),4,16):REM  IZPIS
      I IME PROGRAMA
00350 IF POL-226 THEN POL = POL+32:GOTO 280
00360 IF SP% THEN POL = 2:SN = SN+1:GOTO 230
00370 IF DE = 0 THEN 420
00380 SO = 0:FOP I = NN+1TO NN+DE:PN$(I) = NN
      $:NEXT :PEM "FORDELETEDFILES
00390 PEM  +-----------------------------+
00395 REM  +                             +
00400 REM  +        RAZVRSTITEV          +
00405 REM  +                             +
00410 REM  +-----------------------------+
00420 PRINT "[CLR][14 SPACE]RAZVRSCAM"
00421 PRINT "[11 SPACE]---------------
      [2 DOWN]"
00422 PR$ = "[2 SPACE]CLOSE CLOSE CLOSE
      CLOSE CLOSE CLOSE CLOSE CLOSE CLOSE
      CLOSE CLOSE "
00430 FOR J = 1TO NN-1
00432 :FOP I = J+1TO NN
00434 :IF MID$ (PN$(I),4,16)>MID$ (PN$(J),4,1
      6) THEN 450
00440 :N$ = PN$(I):PN$(I) = PN$(J):PN$(J) = N
      $
00442 :PRINT J;"[HOME][DOWN]":SO = 0
00450 :NEXT I
00451 NEXT J
00452 FOR J = 1TO NN
00454 :PPINT PR$+MID$ (PN$(J),4,16)
00456 :FOR I = 19TO 5STEP -1
00460 :IF MID$ (PN$(J),I,1)<>"[SPACE]" THEN I
       = 5:GOTO 480
```

```
00470 :PN$(J) = LEFT$ (PN$(J),I-1)+CHR$ (160)
      +RIGHT$ (PN$(J),30-I)
00480 :NEXT I
00481 NEXT J
00482 IF SO THEN NEXT :PRINT "[RVS]THE[SPACE]
      END!":END
00500 REM  +-----------------------------+
00501 REM  +                             +
00502 REM  +   NOV ZAPIS DIREKTORIJA     +
00503 REM  +                             +
00504 REM  +-----------------------------+
00520 PN% = 1
00521 FOR J = 0TO SN
00522 :PRINT# 15,"B-P[SPACE]";2;0
00523 REM  ===============================
00524 REM  =                             =
00525 REM  = SLED IN SEKTOR NASL. BLOKA =
00526 REM  =                             =
00527 REM  ===============================
00528 :PRINT# 2,CHR$ (SP(J+1))CHR$ (SE(J+1))
00530 :POL = 2
00531 REM  ===============================
00532 REM  =                             =
00534 REM  =   VPIS PROGRAMOV V BLOK     =
00535 REM  =                             =
00536 REM  ===============================
00537 :FOR I = 1TO 8
00538 :PRINT# 15,"B-P[SPACE]";2;POL
00539 :PRINT# 2,PN$(PN%);
00540 :PN% = PN%+1:REM  NASLEDNJI
00541 :POL = POL+32:REM  PREMIK
00542 :NEXT I
00543 :PRINT# 15,"U2[SPACE]";2;DR%;SP(J);SE(J
      )
00544 NEXT J
00546 CLOSE 15:PEM  ZAPRI ERROR CHANELL
00548 PRINT "[3 DOWN][6 SPACE]DIREKTORIJ
      [SPACE]VPISAN[SPACE]NA[SPACE]DISKETO."
00550 PRINT "[3 DOWN]HAPPY[SPACE]?":END
00560 REM  +-----------------------------+
00561 REM  +                             +
00562 PEM  +   BERI ZNAK Z DISKETE       +
00563 REM  +                             +
00564 REM  +-----------------------------+
00568 GET #2,E$:E$ = CHR$ (ASC (E$+N$)):
      RETURN
00570 IF DS = 0 THEN RETURN   .
00580 PRINT DS$:NEXT :END
00600 PEM  +-----------------------------+
00601 REM  +                             +
00602 REM  +       GLAVA IZPISA          +
00603 REM  +                             +
00604 REM  +-----------------------------+
00610 PRINT "[CLR][YELLOW][13 SPACE]DIREKTORI
      J[SPACE]"
00620 PRINT "[10 SPACE]----------------
      [2 DOWN]"
00630 POKE 53280,0:POKE 53281,0
00690 RETURN
```

Grafični dodatek

COMMODORE 64

Pogostokrat se primeri, da bi želeli svojemu programu, ki smo ga pilili po več tednov skupaj, dodati tudi grafično predstavitev podatkov. Pa najsi gre za izračun in prikaz porabe bencina našega jeklenega konjička ali prikaz porabe električne energije po mesecih.
Vsakdo ve, da številke same ne povedo preveč, sploh pa se zatakne pri primerjavi med posameznimi količinami.
Program, ki nam togo in dolgočasno izpisuje številko za številko, kaj kmalu roma med staro šaro. Čisto nekaj drugega pa je, ko se nam pred očmi prikažejo raznobarvni stolpiči, ki predstavljajo recimo naš osebni dohodek od januarja do maja — sploh šele, če se krivulja strmo dviga
Načinov predstavitev številskih podatkov je seveda ogromno, mi pa si poglejmo, kako to izgleda v basicu. Odločitev je seveda razumljiva, sej imajo ravno ti programerji s takimi in podobnimi problemi največ težav.
Programa samega najbrž nima smisla direktno vključiti v svoj program. Zamišljen je bolj kot prikaz oziroma napotek, kako se lotiti takih stvari na računalniku C-64, kadar uporabljamo basic V2.0 in ne bi radi čakali celo večnost, da nam se v grafiki visoke ločljivosti nejprej zbriše ekran, potem pe počasi nariše ustrezna krivulja.
V začetnih vrsticah si lahko preberete, katere pomembnejše spremenljivke nastdpajo v našem programu. V naslednjih vrsticah pa je pregledno opisan postopek genenranja stolpičev. Spreminjanje barv stolpičem je zelo enostavno, le pri pokih, katerih naslove določajo spremenljivke, ki se začenjajo na FH spremenimo številko berve.
Samo delovanje programa se vidi neposredno iz izpisa, zato si za konec oglejmo še nekaj informacij o rokovanju s programom
Po pravilnem vpisu program poženemo z ukazom RUN Na vprašanje, koliko stolpičev želimo imeti, zadostuje pritisk na ustrezno številko. Možno je predstaviti naenkrat največ 9 stolpičev. Ob pritisku pravilne tipke se nam na ekranu pojavi nov izpis, ki nas povprašuje za vpis velikosti vsakega stolpiča posebej Opozorilo, da je največja vrednost 20, je trebe upoštevati, saj drugače program našega vpisa ne sprejme S prvim pritiskom na številko vpišemo desetico, z drugim pa enico. Pritisk na tipko RETURN ni potreben, saj program sam povpraša po naslednjem vpisu, če je bil vpis prejšnjega pravilen
Po izpisu stolpičev lahko vse skupaj ponovimo še enkrat s pritiskom na tipko F1.

Beti Hočevar

```
00000 REM  +---------------------------+
00001 REM  +         STOLPICI          +
00002 REM  +---------------------------+
00003 REM  + PRIKAZ IZDELAVE GRAFIKONA +
00004 REM  +---------------------------+
00005 REM  +         BETI 1985         +
00006 REM  +---------------------------+
00007 REM
00010 REM  VS() - POLJE VREDNOSTI STOLPICEV
00011 REM  VI   - NASLOV VIDEO RAMA
00012 PEM  BA   - NASLOV BARVNEGA RAMA
00013 REM  V0   - ZACETNI NASLOV RISANJA
00014 REM  B0   - ZACETNI NASLOV BARVANJA
00015 REM  Z    - STEVILO STOLPICEV
00017 REM
00018 PRINT "[CLR]"
00020 POKE 53280,0:POKE 53281,0:REM  EKRAN CR
      NE BARVE
00022 VI = 1024:BA = 55296:REM  NASLOV EKRANA
      IN BARNEGA RAMA
00030 GOSUB 410:REM  VPISI STEVILO STOLPICEV
00110 V0 = VI+920:B0 = BA+920
00114 REM  +---------------------------+
00115 REM  +                           +
```

```
00116 REM  +        NARISI  CRTO            +
00117 REM  +                                +
00118 REM  +--------------------------------+
00120 FOR A = 0TO Z*4-2
00130 :POKE V0+A,99:POKE B0+A,13:REM  CRTA
00140 NEXT A
00144 REM  +--------------------------------+
00145 REM  +                                +
00146 REM  +    RISANJE STOLPICEV           +
00147 REM  +                                +
00148 REM  +--------------------------------+
00150 FOR J = 1TO Z
00160 :IF VS(J) = 0 THEN 340
00164 REM  +--------------------------------+
00165 REM  +                                +
00166 REM  +        NARISI STOLP            +
00167 REM  +                                +
00168 REM  +--------------------------------+
00170 :FOR L = 1TO VS(J)
00180 :BH = V0-L*40:FH = B0-L*40
00184 REM  +--------------------------------+
00186 REM  +      PO DVA KVADRATKA          +
00188 REM  +--------------------------------+
00190 :FOR T = 0TO 2
00200 :POKE BH+T,160:POKE FH+T,5:REM  KVADRAT
      KA V ZELENI BARVI
00210 :NEXT T
00212 :NEXT L
00214 REM  +--------------------------------+
00215 REM  +                                +
00216 REM  +    ZADNJA ZGORNJA CRTA         +
00217 REM  +                                +
00218 REM  +--------------------------------+
00220 :FOR E = 1TO 3
00230 :POKE BH-80+E,100:POKE FH-80+E,14:REM
      "TO SVETLOZELENEBARVE
00232 :NEXT E
00234 REM  +--------------------------------+
00235 REM  +                                +
00236 REM  +   TRIKOTNIK DESNO SPODAJ       +
00237 REM  +                                +
00238 REM  +--------------------------------+
00240 :SS = V0-37:SF = B0-37
00250 :POKE SS,105:POKE SF,6:REM  "STEP ZELEN
      EBARVE
00254 REM  +--------------------------------+
00255 REM  +                                +
00256 REM  +    POTEGNI DESNO STRAN         +
00257 REM  +                                +
00258 REM  +--------------------------------+
00260 :FOR N = 1TO VS(J)-1
00270 :POKE SS-N*40,160:POKE SF-N*40,6
00272 :NEXT N
00280 :POKE SS-VS(J)*40,233:POKE SF-VS(J)*40,
      6
00290 :POKE BH-40,78:POKE FH-40,14:REM  "
00300 :V0 = V0+4:B0 = B0+4
00302 NEXT J
00310 FOR U = 1TO 23
00312 :PRINT
00314 NEXT U:PRINT "[LEFT]";
00320 REM  +--------------------------------+
00321 REM  +                                +
00322 REM  + IZPIS STEVILSKIH VREDNOSTI     +
00323 REM  +                                +
00324 REM  +--------------------------------+
00326 FOR J = 1TO Z
00330 :IF WT(J)>9 THEN PRINT LEFT$ ("[SPACE]"
      +STR$ (VS(J)+"[2 SPACE]",4);
00340 :IF WT(J)<10 THEN PRINT LEFT$ ("[SPACE]
      "+STR$ (VS(J))+"[2 SPACE]",4);
00350 NEXT J
00360 PRINT "[HOME][2 SPACE]PRITISNI[3 SPACE]
      TIPKO[2 SPACE][RVS][2 SPACE]F1[2 SPACE]
      [RVS/OFF][SPACE]ZA[SPACE]KONEC"
00362 GET A$:IF A$<>CHR$ (133) THEN 362
00390 RUN
00400 REM  +--------------------------------+
00401 REM  +                                +
00402 REM  +        VPIS PODATKOV           +
00403 REM  +                                +
00404 PEM  +--------------------------------+
00410 PRINT "[CLR][2 DOWN][8 SPACE]VPISI
      [SPACE]STEVILO[SPACE]STOLPICEV"
00412 PRINT "[DOWN][9 SPACE]NAJVEC[SPACE]JIH
      [SPACE]JE[SPACE]LAHKO[SPACE]9"
00430 GET A$:IF A$<"0" OR A$>"9" THEN 430
00440 Z = VAL (A$)
00450 REM  +--------------------------------+
00451 REM  +                                +
00452 REM  + VPIS VREDNOSTI STOLPICEV       +
00453 REM  +                                +
00454 REM  +--------------------------------+
00455 PRINT "[CLP]"
00456 PRINT "[2 SPACE]VPISI[SPACE]VREDNOSTI
      [SPACE]POSAMEZNIH[SPACE]STOLPICEV"
00457 PRINT "[7 SPACE]NAJVECJA[SPACE]JE
      [SPACE]VREDNOST[SPACE]20"
00458 PRINT "[12 SPACE]STOLPICEV[SPACE]JE"Z
00459 PRINT "[SPACE]--------------------------
      ---------------[DOWN]"
00460 FOR I = 1TO Z
00480 :PPINT "[DOWN]VREDNOST[SPACE]STOLPICA";
      I
00481 :FOR C = 10TO 1STEP -9
00482 ::GET A$:IF A$<"0" OR A$>"9" THEN 482
00484 ::VS(I) = VS(I)+VAL (A$)*C:PRINT "[UP]"
      ,,VS(I)/C;"[3 SPACE]"
00485 ::NEXT C
00486 :IF VS(I)>20 THEN VS(I) = 0:PRINT "[UP]
      ",,"[4 SPACE]":GOTO 481
00494 NEXT I:PRINT "[CLR]"
00499 PETURN
```

Mali računar

Commodore 64

Matematika nam je večini ostala že iz osnovnošolskih dni v spominu kot nekakšen bav-bav. Kakorkoli že — treba se jo je bilo naučiti, to pa je le redkim uspevalo brez trdega dela in vaje. Program MALI RAČUNAR pa bo vsaj tako upam, staršem nadebudnih prvo, drugo in morda celo tretješolčkov prihranil kakšno urico prostega časa več, saj bo »male učenjake« zaposlil z brezkončnim reševanjem računov.

V programu so zajete vse 4 osnovne matematične operacije seštevanje, odštevanje, množenje in deljenje.

Na začetku programa določimo največje število, do katerega bomo računali in operacijo, ki jo želimo vaditi.

Izbira operacije je zelo enostavna, saj na vprašanje odgovorimo z vpisom prve črke ali pa imena izbrane operacije.

Primer: izbira množenja

Vpis: M (RETURN)

ali MNO (RETURN)

ali MNOŽENJE (RETURN)

RETURN v oklepaju pomeni, da moramo pritisniti imenovano tipko.

V kolikor vpis ni pravilen, nas program po njem zopet povpraša.

Sedaj se nam je pojavil osnovni ekran, ki nam poleg računa samega prikazuje trenutno stanje različnih pomembnih podatkov.

Vsakič rešujemo po 20 računov. Številka vprašanja, število pravilnih odgovorov in drugo se nam spreminja ustrezno od reševanja.

Na koncu pa sledi še ocena testa, ki pa je zgolj okvirna.

Program je zasnovan pregledno, tako da ga je enostavno spreminjati in popravljati, ko si bomo zaželeli zahtevnejšega računanja.

Če bi bil začetek za uporabnika prezahteven, bo najbrž pametno popraviti tudi zaključno oceno — zaradi motivacije seveda. Verjetno tudi to ne bo predstavljalo večjega problema.

Beti Hočevar

```
00000  +------------------------------+
00001  +        MALI  RACUNAR         +
00002  +------------------------------+
00003  +   KAKO POSTATI EINSTEIN      +
00004  +------------------------------+
00005  +       JANI APRIL 1984        +
00006  +------------------------------+
00007 GOTO 50
00010  A,B - STEVILI PRI OPERACIJI
00011  C   - REZULTAT
00013  A$  - ZNAK KI GA BEREMO S TIPKOVNICE
00014  O$  - IZBRANA OPERACIJA
00015  C$  - NAS VPIS REZULTATA KOT NIZ
00016  MA  - NAJVECJE MOZNO STEVILO REZULTATA
00017  ZA  - STEVILO PRAVILNIH ODGOVOROV
00018  ZG  - STEVILO NAPACNIH ODGOVOROV
00019  S  - STOLPEC
00020  V  - VRSTICA
00030 POKE 780,0:POKE 781,V:POKE 782,S:SYS 65
      520:RETURN
00050  +----------------------------+
00051  +                            +
00052  +      GLAVNI PROGRAM        +
00053  +                            +
00054  +----------------------------+
00060 GOSUB 800
00062 INPUT "[2 DOWN]NAJVECJE[SPACE]STEVILO:
      [SPACE]";MA
00064 IF MA<20 THEN PRINT "[3 UP]";:GOTO 62
00070 INPUT "[2 DOWN]OPERACIJA:[SPACE]";O$
00080 GOSUB 600:IF O = 0 THEN PRINT "[3 UP]";
      :GOTO 70
00086 GOSUB 700:GOSUB 900:END
00100  +----------------------------+
00101  +                            +
00102  +        SESTEVANJE          +
00103  +                            +
00104  +----------------------------+
00110 GOSUB 510: NAKLJUCNI STEVILI
00114 C = A+B:C$ = ""
00120 IF C>MA THEN 110: POPRAVI A IN B
00128 V = 12:S = 16:GOSUB 30:PRINT "
      [16 SPACE]"
00130 V = 12:S = 16:GOSUB 30:PRINT A"+"B"="
00132 V = 12:S = 27:GOSUB 30
00140 GET A$:IF A$ = "" THEN 140
00142 IF A$ = CHR$ (13) THEN RETURN
00144 IF A$<"0" OR A$>"9" THEN 140
00146 PRINT A$;:C$ = C$+A$:GOTO 140
00200  +----------------------------+
00201  +                            +
00202  +        ODSTEVANJE          +
00203  +                            +
00204  +----------------------------+
00210 GOSUB 510: NAKLJUCNI STEVILI
00214 C = A-B:C$ = ""
00220 IF C>MA OR C<0 THEN 210: POPRAVI A IN B
00228 V = 12:S = 16:GOSUB 30:PRINT "
      [16 SPACE]"
00230 V = 12:S = 16:GOSUB 30:PRINT A"-"B"="
00232 V = 12:S = 27:GOSUB 30
00240 GET A$:IF A$ = "" THEN 240
00242 IF A$ = CHR$ (13) THEN RETURN
00244 IF A$<"0" OR A$>"9" THEN 240
00246 PRINT A$;:C$ = C$+A$:GOTO 240
00300  +----------------------------+
00301  +                            +
00302  +         MNOZENJE           +
00303  +                            +
00304  +----------------------------+
00310 GOSUB 510: NAKLJUCNI STEVILI
00314 C = A*B:C$ = ""
00320 IF C>MA OR C<0 THEN 310: POPRAVI A IN B
00328 V = 12:S = 16:GOSUB 30:PRINT "
      [16 SPACE]"
00330 V = 12:S = 16:GOSUB 30:PRINT A"*"B"="
00332 V = 12:S = 27:GOSUB 30
00340 GET A$:IF A$ = "" THEN 340
00342 IF A$ = CHR$ (13) THEN RETURN
00344 IF A$<"0" OR A$>"9" THEN 340
00346 PRINT A$;:C$ = C$+A$:GOTO 340
00400  +----------------------------+
00401  +                            +
00402  +         DELJENJE           +
00403  +                            +
00404  +----------------------------+
00410 GOSUB 510: NAKLJUCNI STEVILI
00412 B = INT (B/4):IF B = 0 THEN 410: DELJEN
      JE Z NIC
00414 C = INT (A/B):C$ = ""
00420 IF C>MA OR C<0 THEN 410: POPRAVI A IN B
00421 IF A<B THEN 410: REZULTAT JE 0
00428 V = 12:S = 16:GOSUB 30:PRINT "
      [16 SPACE]"
00430 V = 12:S = 16:GOSUB 30:PRINT A"/"B"="
00432 V = 12:S = 27:GOSUB 30
00440 GET A$:IF A$ = "" THEN 440
00442 IF A$ = CHR$ (13) THEN RETURN
00444 IF A$<"0" OR A$>"9" THEN 440
00446 PRINT A$;:C$ = C$+A$:GOTO 440
00500  +----------------------------+
00501  +                            +
00502  +     NAKLJUCNI STEVILI      +
00503  +                            +
00504  +----------------------------+
00510 A = INT (RND (0)*MA+1)
00520 B = INT (RND (1)*MA+1)
00590 RETURN
00600  +----------------------------+
00601  +                            +
00602  +     IZBIRA OPERACIJE       +
00603  +                            +
00604  +----------------------------+
```

```
00610 O = 0
00620 IF LEFT$ ("SESTEVANJE",LEN (O$)) = O$
      THEN O = 1:O$ = "SESTEVANJE"
00630 IF LEFT$ ("ODSTEVANJE",LEN (O$)) = O$
      THEN O = 2:O$ = "ODSTEVANJE"
00640 IF LEFT$ ("MNOZENJE",LEN (O$)) = O$
      THEN O = 3:O$ = "MNOZENJE"
00650 IF LEFT$ ("DELJENJE",LEN (O$)) = O$
      THEN O = 4:O$ = "DELJENJE"
00690 RETURN
00700  +------------------------------+
00701  +                              +
00702  + PONAVLJANJE OPERACIJ         +
00703  +                              +
00704  +------------------------------+
00710 GOSUB 800:GOSUB 840
00712 FOR D = 1TO 20
00720 ON OGOSUB 100,200,300,400
00730 IF C = VAL (C$) THEN ZA = ZA+1
00740 IF C<VAL (C$) THEN ZG = ZG+1
00750 IF C>VAL (C$) THEN ZG = ZG+1: PREMALO
00754 IF C<>VAL (C$) THEN GOSUB 850
00760 GOSUB 840
00780 NEXT
00790 RETURN
00800  +------------------------------+
00801  +                              +
00802  +        PRIPRAVI EKRAN        +
00803  +                              +
00804  +------------------------------+
00810 POKE 53280,14:POKE 53281,14:POKE 646,0
00820 PRINT "[CLR][3 DOWN][6 SPACE]┌──────────
      ──────────────────────┐"
00822 PRINT "[6 SPACE]|[6 SPACE]MALI[SPACE]RA
      CUNALNIK[6 SPACE]|""
00824 PRINT "[6 SPACE]└──────────────────────
      ─────┘"
00830 PRINT "[7 DOWN]........................
      ................"
00839 RETURN
00840 V = 16:S = 2:GOSUB 14:PRINT "NAJVEC
      [SPACE]:[SPACE]";MA
00841 V = 18:S = 2:GOSUB 14:PRINT "OPERACIJA:
      ":PRINT "[DOWN][4 SPACE]";O$
00842 V = 16:S = 22:GOSUB 14:PRINT "POSKUS:
      [3 SPACE]";D+1
00844 V = 18:S = 22:GOSUB 14:PRINT "PRAVILNO:
      [SPACE]";ZA
00846 V = 20:S = 22:GOSUB 14:PRINT "NAPACNO
      [SPACE]:[SPACE]";ZG
00849 RETURN
00850  +------------------------------+
00851  +                              +
00852  +          NAPAKA              +
00853  +                              +
00854  +------------------------------+
00860 V = 24:S = 3:GOSUB 30:PRINT "[RED]NAPAK
      A!!![SPACE]PRAVILNI[SPACE]REZULTAT
      [SPACE]=";C;
00870 GET A$:IF A$ = "" THEN 870
00880 V = 24:S = 3:GOSUB 30:PRINT "[BLACK]
      [35 SPACE]";
00890 RETURN
00900  +------------------------------+
00901  +                              +
00902  +         OCENA                +
00903  +                              +
00904  +------------------------------+
00910 IF ZA<D/2 THEN U$ = "ZELO[SPACE]SLABO":
      GOTO 960
00930 IF ZA<D/1.3 THEN U$ = "SLABO":GOTO 960
00934 IF ZA<D/1.6 THEN U$ = "DOBRO":GOTO 960
00938 IF ZA<D/1.9 THEN U$ = "ZELO[SPACE]DOBRO
      ":GOTO 960
00940 U$ = "ODLICNO"
00960 V = 8:S = 7:GOSUB 14:PRINT "TEST[SPACE]
      JE[SPACE]OPRAVLJEN[SPACE]"+U$
00970 V = 10:S = 10:GOSUB 14:PRINT "ZELIS
      [SPACE]NOV[SPACE]TEST[SPACE][D/N]
      [SPACE]?"
00980 GET A$:IF A$ = "" THEN 980
00990 IF A$ = "D" THEN RUN
00999 END
```

Črte in krogi

SPEKTRUM 48 K

Programček omogoča risanje črt in krogov in je zlasti primeren za cicibane, ki se radi igrajo s spektrumovimi radirkami. Le, da je treba nanje nalepiti etikete, na katere narišemo puščice in druge simbole, ki malčku povedo, kaj se bo zgodilo, če pritisne določeno tipko. Ko se pokaže točka na ekranu s tipkami E, R, T, D, F, X, C, V, spreminjaš njen položaj in jo premikaš po ekranu. Če pritisneš še U, bo sled ostala vidna. Krog narišeč tako, da točko »odpelješ« v središče, pritisneš tipko N in jo nato »odpelješ« nazaj na rob. Krog se izriše le v primeru, če je manjši od ekrana. S tipko P zbrišeš sliko in očistiš ekran, O pa vklopi oziroma izklopi zvočni signal. Tipka G je za snemanje slike na kaseto, s številkami od 0 do 7 pa lahko spreminjaš barvo črte. Z uporabljaš za kopiranje slike na tiskalnik. To je vsa umetnost. Veliko zabave!

```
   10 REM  risanje po ekranu
   20 REM  primerno za otroke, ki
 se ne poznajo crk in stevilk
   30 CLS : LET x=127: LET y=87: L
ET z=1: LET r=0: LET tocka=1: LET
 bip=0: LET ink=0
   33 PRINT #1;"Pritisni"; INVERSE
 1;"ERTDFXCVUIOPGJZNMO-7"
   35 LET s$=INKEY$: IF s$="" THEN
   GO TO 35
   40 IF s$="r" THEN  LET dx=0: LE
T dy=1: GO TO 200
   42 IF s$="t" THEN  LET dx=1: LE
T dy=1: GO TO 200
   45 IF s$="f" THEN  LET dx=1: LE
T dy=0: GO TO 200
   47 IF s$="v" THEN  LET dx=1: LE
T dy=-1: GO TO 200
   50 IF s$="c" THEN  LET dx=0: LE
T dy=-1: GO TO 200
   52 IF s$="x" THEN  LET dx=-1: L
ET dy=-1: GO TO 200
   54 IF s$="d" THEN  LET dx=-1: L
ET dy=0: GO TO 200
   56 IF s$="e" THEN  LET dx=-1: L
ET dy=1: GO TO 200
   58 IF s$="u" THEN  LET z=0: LET
 r=0: GO TO 75
```

```
60 IF s$="i" THEN LET z=1: LET r=0: GO TO 35
62 IF s$="j" THEN LET z=0: LET r=1: GO TO 35
64 IF s$="g" THEN INPUT "ime?";l$: SAVE l$SCREEN$ : GO TO 35
66 IF s$="p" THEN GO TO 30
68 IF s$="o" THEN LET bip=NOT bip: LET vis=44: GO TO 35
70 IF s$>="0" AND s$<="7" THEN LET ink=VAL s$: GO TO 35
72 IF s$="n" THEN LET xs=x: LET ys=y: GO TO 35
80 IF s$<>"m" THEN GO TO 140
90 LET rad=SQR ((x-xs)*(x-xs)+(y-ys)*(y-ys)): IF rad=0 THEN GO TO 35
95 IF xs+rad>254 THEN GO TO 35
97 IF xs-rad<1 THEN GO TO 35
100 IF ys+rad>174 THEN GO TO 35
110 IF ys-rad<1 THEN GO TO 35
120 CIRCLE INK ink; OVER r;xs,ys,rad
140 IF s$="z" THEN COPY : GO TO 35
145 GO TO 35
200 IF x+dx>254 THEN GO TO 300
210 IF x+dx<1 THEN GO TO 300
230 IF y+dy>174 THEN GO TO 300
235 IF y+dy<1 THEN GO TO 300
240 IF r THEN PLOT OVER 1;x,y
250 IF z AND NOT tocka THEN PLOT OVER 1;x,y
260 LET x=x+dx: LET y=y+dy
270 LET tocka=POINT (x,y)
300 PLOT INK ink;x,y
310 IF bip THEN BEEP 0.012,vis
320 GO TO 35
```

»Osvežitev« že znane igre

SPECTRUM 48 K

Igra, ki jo danes predstavljamo, vam je gotovo znana. Že nekaj časa jo spremljate v kvizu TV Ljubljana, najbolj navdušeni pa se boste podobne spomnili tudi iz priročnika, ki ste ga dobili z vašo Mavrico »Obešenca« ali »Vislice«, kakor igro običajno imenujemo, so včasih igrali s papirjem in svinčnikom, a lahko se je lotimo z računalnikom. Gre za to, da si eden od igralcev (ali — v našem primeru — tudi računalnik) zamisli besedo, ki jo mora soigralec nato uganiti. Ugibati mora po črkah, če se zmoti več kot sedemkrat, je izgubil — pravimo, da je »obešen«. Igra je prav zabavna, če pa se je loti večje število igralcev, včasih kar razburljiva. Vendar: večkrat smo že omenili, da nam v Zelenem Bitu ne gre le za same objavljene programe. Predstaviti želimo zlasti nekatere prijeme, ki bi vam koristili v lastnih programih. To velja tudi za »Vislice«. Progam je kratek, zato se bomo nekaterih podrobnosti dotaknili kasneje, v opombah. Tu bomo pojasnili le glavni »trik«, prijem, ki se nam zdi najpomembnejši.

Kot vemo, na Spectrumu ni nekaterih naših črk. Deloma jih sicer nadomestimo z uporabniškimi grafičnimi znaki, vendar je to precej zasilen izhod. Kaj neprijetno je namreč, če morate pritisniti tri tipke, da bi dobili eno črko. Prav tako neprijetno pa je, če se našim šumnikom odrečemo in je potem koža koza, žima zima in črka crka. In vendar je rešitev težav preprosta! »Prevarati« moramo računalnik, tako da bo napisal č, kadar bomo pritisnili npr. Symbol shift in a (seveda bi lahko izbrali tudi katerokoli drugo kombinacijo). V ta namen si moramo pomagati s strojnim jezikom. Ne ustrašite se! Vse je pripravljeno in uporabno tudi za tistega, ki za strojni jezik še slišal ni (za poznavalce pa smo na koncu dodali tudi program v zbirniku). Naš strojni programček (shranjen v vrsticah 2040 in 2050) povzroči, da Mavrica izpiše č, š oziroma ž, kadar pritisnemo Symbol shift ter a, s oziroma d (običajno bi dobili STOP, NOT oz. STEP). Pričujoči program omogoča le izpis malih črk, ker velikih v Vislicah ne bomo potrebovali (in tudi DATA stavki bi se nekoliko razvlekli, brez napora pa bi lahko vključili tudi velike. Program je uporaben tako na Spectrumu 16k kot 48k, vendar je treba za 16k v vrstici 3000 pri ukazu POKE USR »a«-88, 255 nadomestiti 255 s 127, v vrstici 2050 pa vrednost »254« (šesta od konca) s »126«. Ko boste želeli naš dodatek uporabiti v svojih programih, boste vanje prenesli vrste od 2040 do 2070 ter vrsto 3000. Seveda jim lahko spremenite številke (morda najbolje, če nosijo številke od 9994 do 9998), ne pozabite pa ob tem spremeniti vrednost RESTORE v vrsti 3000. A da ne bo nesporazuma! Ko boste tipkali program, bo še vedno treba rabiti uporabniške grafične znake (tudi v »Obešencu«). Izboljšavo bo moč uživati, ko bomo program pognali.

Na začetku smo rekli, da lahko besede za ugibanje zastavi računalnik sam. Vendar se jih mora prej »naučiti«. Najlaže bo šlo takole: pretipkajte priloženi program in ga shranite na trak s SAVE »Vislice« LINE 3000. Nato ga poženite z RUN 3000, počakajte približno 10 sekund in pritisnite BREAK. Dodajte vrstico

9000 FOR i = 1 TO 80: INPUT (i), a$: NEXT i

in jo poženite. Vnesli boste besede ter jih shranili s SAVE »besede« DATA a$(). To je vse. Trenutno ima program prostor za 80 besed, dolgih največ 10 znakov. Ti dve vrednosti lahko spremenite, a morate ustrezno spremeniti tudi vrste 50, 60 in 3010 (ter 9000, kadar je). Lahko si pripravite več zbirk besed, če bodo »Vislice« pogosto v rabi. In naj vas pomirim! Kar slišim namreč, kako se nekateri pritožujete: »Kaj bom ugibal besede, ki sem jih sam zapisal!« Tudi sam bi mislil tako, ko ne bi programa z zbirko besed vred za nekaj časa postavil »na hladno«. Ko sem ga potem spet pognal, pa sem kar nekajkrat »zabingljal«.

Zdaj pa kar k programu, za lažje razumevanje smo nanizali še opombe.

3000 — 3010

Začetek programa. Prebere se strojni program in grafični znaki. Naloži se zbirka besed (imejte jo posneto kar za samim programom). Polje b$, nam bo služilo za kontrolo (gl. spodaj).

3020 — 3070

Navodila. Če so vam odveč, jih lahko izpustite, a saj veste: programi brez navodil so kot bicikel brez sedeža. V vrsti 3020 vidite »...PRINT ≠o;...« Ta način vam omogoča pisanje v 23. in 24. vrstico zaslona.

10 — 30

Odločimo se ali bo skrito besedo izbiral računalnik. Izris možica na zaslon.

40

Besedo bo vpisoval nekdo od igralcev. Naj vas opozorim na »prikrivanje«, ki ga lahko uporabite, kadar ne želite, da bi bilo vidno besedilo v spodnjem delu (23. in 24. vrsta) zaslona. Vpišite PRINT PEEK 23624. Rezultat (denimo, da je bil 56) delite z 8 (mi dobimo 7) in to prištejte k prvemu rezultatu (kar nam da 63). To vrednost naložimo nazaj (POKE 23624, 63) in zaščita je vključena. Izključimo jo s POKE 23624, prvi rezultat (56 v našem primeru).

50 — 80

Besedo izbira računalnik (s pomočjo naključnih števil). Ker je naključno število lahko 0, ne more pa biti 1, ne smemo zapisati LET z = 90*RND (to bi utegnilo pripeljati do napake). B$ nam služi za kontrolo, ali je bila izbrana beseda v igri že zastavljena. Če je bila, je treba izbrati drugo.

100 — 160

Ugibanje črke. Prve tri črke pri ugibanju morajo biti soglasniki, kar nadzira vrstica 110.

170

Je igralec že uganil besedo? Če je, pojdi na 220.

190

Če se je igralec že 8 × zmotil, obesi možica.

200 — 210

Črka ni prava. Nariši naslednji del vislic.

220 — 290

Beseda je razkrita! Nariši veselega možica, ki je ušel izpod vešal, ter vprašaj za ponovitev igre.

300 — 380

Igralcu ni uspelo. Obesi možica ter vprašaj za ponovitev igre.

1000 — 1040

Podprogram, ki nariše možica na položaju × (× je treba določiti v glavnem programu, pred klicem podprograma — gl. npr. vrstico 30).

2000 — 2030

Podatki o posameznih delih vešal.

2040 — 2050

Strojni program.

2060

Točkovni vzorci za č, š, ž (male črke).

2070

Točkovni vzorci za Č, Š, Ž (velike črke).

Opozorilo: namesto znakov, ki so v našem izpisu podčrtani, uporabljajte pri tipkanju programa uporabniške grafične znake (dosežete jih s pritiskom na CAPS SHIFT in 9 ter ustrezno črko).

Jaro Lajovic

VISLICE — Program v zbirniku

```
        RST   56
        EXX
        PUSH  AF
        LD    A,(23560)
        CP    194
        JR    C,KONEC
        LD    B,144
        CP    226
        JR    Z,ZAMEN
        INC   B
        CP    195
        JR    Z,ZAMEN
        INC   B
        CP    205
        JR    NZ,KONEC
ZAMEN   LD    A,B
        LD    (23560),A
KONEC   POP   AF
        EXX
        RET
VKLOP   LD    A,254
        LD    I,A
        IM    2
        RET
```

```
10 RESTORE : LET r=0: PRINT #0; AT 1,0;"Izbira besedo raAunalnik ?
     (d/n)": IF INKEY$ ="d" THEN LET r=1: GO TO 30
20 IF INKEY$ <> "n" THEN GO TO 10
30 BORDER 5: CLS : RANDOMIZE : LET x=240: GO SUB 1000: PLOT 238,128:
     DRAW 4,0
40 IF NOT r THEN POKE 23624,45: INPUT LINE w$: POKE 23624,40:
     LET b= LEN w$: LET v$=z$( TO b): GO TO 80
50 LET z=1+79* RND : IF b$(z) <> " " THEN GO TO 50
60 LET b$(z)="1": FOR i=10 TO 1 STEP -1: IF a$(z,i)=" " THEN NEXT i
70 LET b=i: LET v$=z$( TO b): LET w$=a$(z, TO b)
80 LET c=1: LET d= NOT c: LET ct=c
90 FOR n=1 TO b: PRINT AT 20,n;"-";: NEXT n
100 INPUT "ugani Arko: "; LINE g$: IF g$="" THEN GO TO 100
110 LET g$=g$(1): IF ct<4 AND (g$="a" OR g$="e" OR g$="i" OR g$="o"
     OR g$="u") THEN GO TO 100
120 LET ct=ct+1: PRINT AT 1,c;g$
130 LET c=c+1: LET u$=v$
140 FOR n=1 TO b: IF w$(n)=g$ THEN LET v$(n)=g$
150 NEXT n
160 PRINT AT 19,1;v$
170 IF v$=w$ THEN GO TO 220
180 IF v$ <> u$ THEN BEEP .3,0: BEEP .3,4: BEEP .5,7: GO TO 100
190 IF d=8 THEN GO TO 300
200 BEEP .3,10: BEEP .3,9: BEEP .3,8: LET d=d+1
210 READ x0,y0,x,y: PLOT x0,y0: DRAW x,y: GO TO 100
220 FOR i=1 TO 3: BEEP .1,0: BEEP .1,4: BEEP .1,7: NEXT i:
     BEEP .3,-5: BEEP .4,-8: BEEP .6,-12
230 OVER 1: LET x=240: GO SUB 1000
240 PLOT 238,128: DRAW 4,0
250 OVER 0: LET x=146: GO SUB 1000
260 PLOT 143,129: DRAW 6,0, PI /2
270 PRINT #0; AT 1,6;"Ee enkrat ? (d/n)": IF INKEY$ ="n" THEN GO TO
     9999
280 IF INKEY$ <> "d" THEN GO TO 270
290 RESTORE : GO TO 30
```

```
 300 OVER 1: PLOT 255,65: DRAW -48,0: DRAW 0,-48
 310 PLOT 238,128: DRAW 4,0
 320 PLOT 255,117: DRAW -15,-15: DRAW -15,15
 330 OVER 0: PLOT 236,81: DRAW 4,21: DRAW 4,-21
 340 OVER 1: PLOT 255,66: DRAW -15,15: DRAW -15,-15
 350 OVER 0: PLOT 236,60: DRAW 4,21: DRAW 4,-21
 360 PLOT 237,127: DRAW 6,0,- PI /2
 370 FOR i=10 TO -16 STEP -1: BEEP .03,i: NEXT i: BEEP .3,-16:
      BEEP .3,-17: BEEP .5,-18
 380 PRINT AT 19,1;w$: GO TO 270
1000 CIRCLE x,132,8: PLOT x+4,134: PLOT x-4,134: PLOT x,131
1010 PLOT x,123: DRAW 0,-20: PLOT x,101: DRAW 0,-19
1020 PLOT x-15,66: DRAW 15,15: DRAW 15,-15
1030 PLOT x-15,117: DRAW 15,-15: DRAW 15,15
1040 RETURN
2000 DATA 120,65,135,0,184,65,0,91
2010 DATA 168,65,16,16,184,81,16,-16
2020 DATA 184,156,68,0,184,140,16,16
2030 DATA 204,156,-20,-20,240,156,0,-16
2040 DATA 255,217,245,58,8,92,254,194,56,20,6,144,254,226,40,10,4,254,
      195,40
2050 DATA 5,4,254,205,32,4,120,50,8,92,241,217,201,62,254,237,71,237,9
      4,201
2060 DATA 20,8,28,32,32,32,28,0,20,8,28,32,28,2,60,0,40,16,124,8,16,32
      ,124,0
2070 DATA 24,60,66,64,64,66,60,0,24,60,64,60,2,66,60,0,24,126,4,8,16,3
      2,126,0
3000 CLEAR USR "a"-90: RESTORE 2040: FOR i= USR "a"-40 TO USR "g"-1:
      READ r: POKE i,r: NEXT i: RESTORE: POKE USR "a"-89,48:
      POKE USR "a"-88,255: LET r= USR (USR "a"-7): CLS
3010 DIM a$(80,10): DIM b$(80): LOAD "" DATA a$(): INK 0: PAPER 7: CLS:
      PRINT AT 10,10;"Trenutek ...":
      LET z$="                                ": REM 32 presledkov
3020 PRINT #0; AT 1,7;"Navodila ? (d/n)": IF INKEY$ ="n" THEN CLS :
      GO TO 10
3030 IF INKEY$ <> "d" THEN GO TO 3020
3040 CLS : BORDER 3: PRINT '"            VISLICE"''"  Pri  tej  znani
       igri  moraB  uganiti  skrito  besedo, ki jo  lahko zastavi
      eden od igralcev  ali pa raAunalnik.  Program te  na
      zaAetku vpraBa, katero moC-  nost  CeliB.  Pri  prvi
      moraB  vpisati besedo,  ki  pa ostane  med vpisovanjem
      skrita."
```

```
3050 PRINT "  Drke  A, B in C dobiB s pri-  tiskom na SYMBOL
       SHIFT ter"'' TAB 11;"a za A,"' TAB 11;"s za B in"' TAB
       11;"d za C."''
3060 PRINT "  Pri  ugibanju  prve tri Arke  ne smejo biti
       samoglasniki."
3070 PRINT #0; AT 0,0;z$;"  Pritisni katerokoli tipko    ": PAUSE 0:
       CLS : GO TO 10
```

Skakač za hitro reagiranje

TI-99/4A

Če ste lastnik računalnika TI-99/4A gotovo že dolgo pogrešate programe zanj. Program Skakač je igra, pri kateri je najbolj pomembno hitro reagiranje. Stopiti moramo na vse stopnice (enkrat ali večkrat odvisno od nivoja), pri tem pa moramo paziti še na »pošast«. Če pritisneš na »space«, si za nekaj časa nesmrten, lahko pa tudi ujameš pošast (150 točk). Skakača vodiš s tipkami: A (levo gor), X (desno dol), L (desno gor), M (levo dol). Navodila so tudi v programu. Ker smo program »printali« na printerju Commodorja, ni bilo moč natisniti malih črk. Namesto njih smo tiskali reverzno. Torej, črke, ki so v listingu pisane reverzno, vtipkajte v vaš TI-99/4A kot male. Veliko uspeha pri igranju!

Tomaž Ziherl

```
100 REM *******************
110 REM *      SKAKAC     *
120 REM *******************
130 REM
140 REM *******************
150 PEM *ZA TEXAS TI-99/4A*
160 PEM *******************
170 REM
180 REM *******************
190 PEM *      @ 1985     *
200 REM *******************
210 REM
220 PEM DEFINICIJA CHAR
230 CALL CLEAP
240 FOR Q=128 TO 152 STEP 8
250 CALL CHAR(Q,"00FFFFFF")
260 CALL CHAR(Q+1,"FFFFFFFFFFFFFFFF")
270 NEXT Q
280 CALL CHAR(112,"7C547CD66C3828EE")
290 CALL CHAR(96,"7CB6FEAA82AA7C")
300 CALL CHAR(138,"0000B7A5B595B7")
310 CALL COLOR(12,10,1)
320 CALL COLOR(13,16,1)
330 CALL COLOR(14,15,1)
340 CALL COLOR(15,14,1)
350 CALL COLOR(15,13,1)
360 FOR Q=1 TO 11
370 CALL COLOP(Q,12,1)
380 NEXT Q
390 REM NAVODILA
400 CALL SCREEN(2)
410 PRINT TAB(8),"▒▒▒SKAKAC▒▒▒"::::
420 PPINT "S SKAKACEM SKACES S PRITI-":"SKOM NA TIPKE: A,X,L,M. ZA   "
425 PRINT "LIKVIDACIJO POSASTI PA":
430 PPINT "PPITISNI NA 'SPACE' IN":"UJAMI POSAST, VENDAR SAMO,  ":
```

```
435 PRINT "  DOKLER IGRA MELODIJA!"::"BISTVO IGRE JE V":
440 PRINT " TEM, DA":"STOPIS NA VSE 'STOPNICE',":"KI ZATEM SPREMENIJO BARVO"::
445 PRINT " VELIKO SRECE V IGRANJU!"
450 PRINT "■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■"
460 REM
470 PRINT :"PRITISNI TIPKO ZA ZACETEK!"
480 CALL KEY(0,K,S)
490 IF S=0 THEN 480
500 V=5
510 P=0
520 R=0
530 S=1
540 BP=3
550 REM     **SLIKA**
560 CALL CLEAR
570 FOR Q=1 TO 7
580 FOR W=1 TO Q
590 CALL HCHAR(Q*2+4,13-Q*2+W*4,128,3)
600 NEXT W
610 NEXT Q
620 E=(INT((S-1)/3)+1)*8+129
630 MR=(INT((S-1)/3)+1)*28
640 CALL HCHAR(24,1,E,32)
650 IF V<5 THEN 790
660 O$="0204■ ■ ■ ■"
670 GOSUB 730
680 O$="0220TOCKE:"&STR$(P)
690 GOSUB 730
700 CALL HCHAR(4,4,42,BP)
710 GOTO 790
720 REM       **PRINT AT**
730 Q=VAL(SEG$(O$,1,2))
740 W=VAL(SEG$(O$,3,2))
750 FOR Z=5 TO LEN(O$)
760 CALL HCHAR(Q,W-5+Z,ASC(SEG$(O$,Z,1)))
770 NEXT Z
780 RETURN
790 A=5
800 B=16
810 X=19
820 Y=30
830 O$="2107BODI PRIPRAVLJEN"
840 GOSUB 730
850 FOR Q=1 TO 500
860 NEXT Q
870 CALL HCHAR(21,1,32,32)
880 CALL HCHAR(A,B,112)
890 REM       **PREMIKANJE SKAKACA**
900 CALL KEY(0,K,W)
910 IF K<>32 THEN 980
920 IF (BP=0)+(PP>0)THEN 1430
930 BP=BP-1
940 CALL HCHAR(4,4+BP,32)
950 CALL CHAR(96,"7CB6FEC682BA7C")
960 PP=1
970 GOTO 1430
980 I=(K=65)+(K=76)-(K=88)-(K=77)
990 J=(K=65)+(K=77)-(K=88)-(K=76)
1000 IF (I=0)+(J=0)THEN 1430
1010 CALL HCHAR(A,B,32)
1020 A=A+I*2
1030 B=B+J*2
1040 CALL HCHAR(A,B,112)
1050 IF (A<>X)+(B<>Y)THEN 1190
1060 IF PP=0 THEN 1680
1070 CALL HCHAR(A,B,138)
1080 P=P+150
1090 CALL SOUND(100,330,3,331,3)
1100 CALL SOUND(100,294,3,295,3)
1110 CALL SOUND(100,262,3,263,3)
1120 CALL SOUND(250,330,3,331,3)
1130 O$="0226"&STR$(P)
1140 GOSUB 730
1150 CALL HCHAR(A,B,112)
1160 X=19
1170 Y=30
1180 CALL HCHAR(X,Y,96)
1190 CALL GCHAR(A+1,B,G)
1200 IF G=32 THEN 1680
1210 IF G=E-1 THEN 1430
1220 CALL SOUND(100,G,1)
1230 CALL HCHAR(A+1,B-1,G+8,3)
1240 R=R+1
1250 P=P+20
1260 IF R<MR THEN 1430
1270 S=S+1
1280 IF S<10 THEN 1300
1290 S=9
1300 R=0
1310 FOR K=1 TO BP
1320 CALL SOUND(200,856,1,855,1,854,1)
1330 CALL HCHAR(4,BP-K+4,32)
1340 P=P+50
1350 O$="0226"&STR$(P)
1360 GOSUB 730
1370 NEXT K
1380 BP=BP+1
1390 CALL HCHAR(4,4,42,BP)
1400 CALL HCHAR(A,B,32)
1410 CALL HCHAR(X,Y,32)
1420 GOTO 570
1430 REM        **PREMIKANJE**
1440 IF PP=0 THEN 1500
1450 PR=PR+1
1460 CALL SOUND(-300,1760-PR*110,1)
1470 IF PR<15 THEN 880
1480 PR=0
1490 CALL CHAR(96,"7CB6FEAA82AA7C")
1500 CALL HCHAR(X,Y,32)
1510 IF INT(RND*10)<=(S-INT(S/3)*3)THEN 1560
1520 I=SGN(RND-.5)
1530 J=SGN(RND-.5)
1540 CALL GCHAR(X+I*2+1,Y+J*2,G)
1550 IF G<>32 THEN 1580
1560 I=SGN(A-X)
1570 J=SGN(B-Y)
1580 IF (I<>0)*(J<>0)THEN 1630
1590 IF J=0 THEN 1620
1600 I=-1
1610 GOTO 1630
1620 J=SGN(RND-.5)
1630 X=X+I*2
1640 Y=Y+J*2
1650 CALL HCHAR(X,Y,96)
```

```
1660 IF (A=X)*(B=Y)THEN 1680
1670 GOTO 900
1680 REM        **MRTEV SKAKAC**
1690 CALL SOUND(-1000,-6,0,110,1)
1700 V=V-1
1710 IF V=0 THEN 1780
1720 CALL HCHAR(2,V*2+2,32)
1730 CALL HCHAR(A,B,32)
1740 CALL HCHAR(X,Y,32)
1750 Q$="0226"&STR$(P)
1760 GOSUB 730
1770 GOTO 790
1780 REM        **KONEC IGRE**
1790 Q$="0226"&STP$(P)
1800 GOSUB 730
1810 Q$="2007G A M E  O V E R"
1820 GOSUB 730
1830 Q$="2208ZELIS SE[Y/N]?"
1840 GOSUB 730
1850 CALL KEY(0,K,S)
1860 IF K=89 THEN 500
1870 IF K<>78 THEN 1850
1880 CALL CLEAR
1890 END

READY.
```

POZOR!!

W=p

p-ji, ki so v listingu pisani reverzno, so v texas basicu pisani malo.